DOCUMENT RESUME

ED 207 802 SE 035 492

AUTHOR	Watt, Daniel
TITLE	Final Report of the Brookline LOGO Project, Part III: Profiles of Individual Student's Work. Artificial Intelligence Memo No. 546.
INSTITUTION	Massachusetts Inst. of Tech., Cambridge. Artificial Intelligence Lab.
SPONS AGENCY	National Science Foundation, Washington, D.C.
REPORT NO	LOGO-54
PUB DATE	Sep 79
GRANT	NSF-77-19083-SEI
NOTE	224p.; For related document, see SE 035 491.
AVAILABLE FROM	Artificial Intelligence Lab., 545 Technology Square, Rm. 338, Cambridge, MA 02139 ($4.00).
EDRS PRICE	MF01/PC09 Plus Postage.
DESCRIPTORS	Biographical Inventories; *Case Studies; Computer Programs; *Computer Science Education; Elementary Education; Elementary School Mathematics; *Grade 6; *Learning Activities; Mathematics Education; Mathematics Instruction; Profiles; *Programing; Student Developed Materials
IDENTIFIERS	*LOGO Programing Language; *Mathematics Education Research

ABSTRACT

During the school year 1977-78, four computers equipped with LOGO Turtle Graphics were installed in an elementary school in Brookline, Massachusetts. All sixth-grade students in the school had between 20 and 40 hours of hands-on experience with the computers. The work of 16 students was documented in detail. The profiles, written by the classroom teacher, are discursive essays on the experiences of each of the sixteen experimental subjects. This illustrates the wide variety of learning styles and learning paths within the LOGO learning environment. They are particularly useful for teachers who anticipate using LOGO with children, and offer a rich source of project ideas suitable for naive programers. (Author)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED207802

SE

Massachusetts Institute of Technology

Artificial Intelligence Laboratory

A.I. Memo No. 546

September 1979
LOGO Memo No. 54

FINAL REPORT OF THE BROOKLINE LOGO PROJECT
PART III: Profiles of Individual Student's Work.
Daniel Watt

Abstract

During the school year 1977/78 four computers equipped with LOGO and Turtle Graphics were installed in an elementary school in Brookline, Mass. All sixth grade students in the school had between 20 and 40 hours of hands-on experience with the computers. The work of 16 students was documented in detail.

The profiles, written by the classroom teacher, are discursive essays on the experiences of each of the sixteen experimental subjects. This illustrates the wide variety of learning styles and learning paths within the LOGO learning environment. They are particularly useful for teachers who anticipate using LOGO with children, and offer a rich source of project ideas suitable for naive programmers. (See LOGO Memo 53 for Part II of this report.)

The work reported in this paper was supported in part by the National Science Foundation under grant number 77-19083SED and conducted at the Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts. The views and conclusions contained in this paper are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the National Science Foundation or the United States Government.

SE 035 492

TABLE OF CONTENTS

1. Albert
2. Betsy
3. Darlene
4. Deborah
5. Dennis
6. Donald
7. Gary
8. Harriet
9. Jimmy
10. Karl
11. Kathy
12. Kevin
13. Laura
14. Monica
15. Ray
16. Tina

INTRODUCTION

In this part of our report we present sixteen separate profiles, describing the LOGO experiences of each of our subjects in some detail. Each profile includes a statement of how the child is perceived as a student in the regular academic areas of the school, a description of "what the child learned" in the LOGO classes, an analysis of each child's particular strengths and problems, and the particular teaching strategies that were considered appropriate for each child. In reading this material, one should bear in mind that the students' learning took place in a project-oriented setting and no attempt was made to expose all students to a "standard LOGO curriculum." Rather, the teacher introduced new material to students on an individual basis, and in a way which would be integrated into their individual projects. Consequently, we observed different students concentrating on different aspects of LOGO. For example, some organized most of their learning experiences around the creation of free-form "emergent" designs, while others concentrated on elaborately planned projects. Most of the students' work related to computer graphics, but a few also undertook non-graphics projects. The sixteen students in our experimental sample spanned a wide range of interests, abilities and cognitive styles. One of the strengths of this kind of LOGO learning environment is that it appealed to students across such a spectrum and provided a significant learning experience for each of them.

1. Albert

Albert is considered to be an average student by his teachers. (His most recent scores on the standardized tests given to all sixth graders place him in the 51st percentile nationally.) He is a quiet, well mannered student, serious in his work, and well liked by his classmates. When asked questions about himself or asked to explain his thinking, he tends to respond with brief statements, or to be non-committal.

Albert thoroughly enjoyed working with the computer, and became deeply involved with computer activities at the very first class. He rarely seemed to be at a loss or to need help in choosing an activity. He was adept at moving the TURTLE around the screen, and using it to draw simple pictures. He usually worked on his own, rarely asking for help, and often worked on a number of different small tasks during the course of a single class period. Although he did tackle a couple of longer animation projects, he seemed most comfortable with shorter tasks that he could carry out without assistance. Albert concentrated on his own activities and seemed to be largely unaware of the work of other students.

Although Albert was successful in driving the TURTLE and in carrying out many small projects, he had difficulty using subprocedures and variables, and in planning his work more than one step at a time. Within these limitations, he himself was extremely satisfied with his work, and expressed a strong interest in continuing to work with computers next year.

1.1 Albert's work in Turtle Geometry

Albert began his LOGO experience by "driving the TURTLE" with great proficiency. In his first session of LOGO, Albert was able to use PENUP and PENDOWN to try out steps experimentally. He readily used FORWARD/BACK, and RIGHT/LEFT as inverses, estimated distances and directions, and aggregated two forward steps into one (combining FORWARD 35 and FORWARD 23 into one step by typing FORWARD 58). He made expert use of 90 degree rotations to move the TURTLE into parallel positions, and at one point used a 45 degree turn to create a diagonal line.

In the third class he drew a complex "face" using direct commands, and spent the next two classes writing and debugging a lengthy procedure which drew the face. Although the teacher suggested the use of subprocedures, Albert rejected the idea, preferring to work with a long sequential procedure.

```
TO KEITH
1 RCIRCLE 89
2 PENUP
3 RIGHT 90
4 FORWARD 45
5 LEFT 90
6 FORWARD 23
7 PENDOWN
8 RCIRCLE 12
9 PENUP
10 RIGHT 90
11 FORWARD 55
12 FORWARD 11
13 BACK 9
14 BACK 5
15 RIGHT 90
16 FORWARD 6
17 PENDOWN
18 LCIRCLE 12
19 PENUP
20 FORWARD 15
21 PENDOWN
22 RCIRCLE 13
23 PENUP
25 FORWARD 45
26 RIGHT 90
27 FORWARD 50
28 PENDOWN
29 BACK 65
30 BACK 23
31 PENUP
32 RIGHT 90
33 FORWARD 100
34 FORWARD 23
35 LEFT 90
36 FORWARD 45
37 RIGHT 90
38 FORWARD 6
END
```

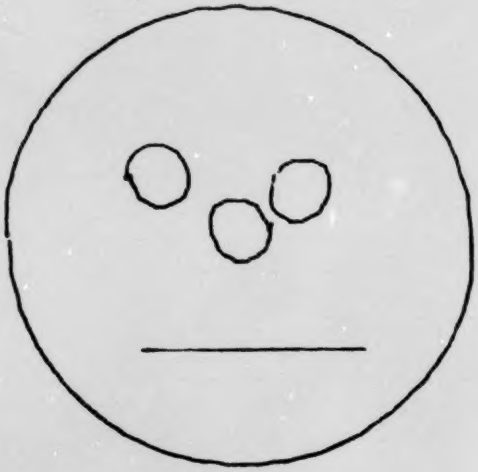


Figure 1.1

In drawing his face, Albert used 90 degree turns to "drive the TURTLE" to different locations on the screen. This enabled him to keep the different features of the face parallel to each other. On the other hand, the repeated use of 90 degree turns and small TURTLE steps, made it difficult for Albert to consolidate his steps, or to understand the purpose of each step or series of steps in his procedure. He drew his design by trial and error, and then incorporated all the steps into a procedure by copying them into his notebook. He did not copy exploratory steps which were carried out with PENUP. Since some PENUP steps were needed as part of his drawing, and others were exploratory, and not needed in the final design, Albert experienced some confusion in figuring out which steps to include and which to eliminate. This led to bugs in his procedure, which were difficult for him to understand.

Albert also had difficulty deciding whether to use a circle that curved to the right or to the left for the interior features of his face. He could get the TURTLE to the point where he wanted the eye drawn, but seemed unable to predict consistently whether the LCIRCLE or RCIRCLE command was needed to place the circle where he wanted it. He would first try the circle with PENUP, then repeat it with PENDOWN if it was right, or choose the reverse circle if it came out in the wrong place. This created more possibilities for copying errors and further complicated the process of debugging the final procedure.

In the sessions that followed, Albert continued to move the TURTLE around the screen by creating a kind of "grid" composed of 90 degree turns. Albert made use of angles other than 90 degrees only when they were specifically needed for particular shapes.

In the sixth class, Albert decided to make a five pointed star. After several unsuccessful tries his teacher suggested that he use the REPEAT command, repeating a forward step and a turn five times, and then varying the angle to find the correct rotation. Starting at 60 degrees and using a method of successive approximations, it took him seven tries to find the correct angle: he tried 60, 110, 150, 140, 145, 143 and finally was successful with 144. For the last three steps, the teacher suggested that he hide the TURTLE each time, to see if the star was "exactly right".

When Albert wrote his procedure TO STAR, he listed 10 separate steps, rather than using the REPEAT command he had used in his explorations. This was another indication of his preference for linear, sequential procedures, rather than subprocedures.

```
TO STAR
1 FORWARD 100
2 RIGHT 144
3 FORWARD 100
4 RIGHT 144
5 FORWARD 100
6 RIGHT 144
7 FORWARD 100
8 RIGHT 144
9 FORWARD 100
10 RIGHT 144
END
```

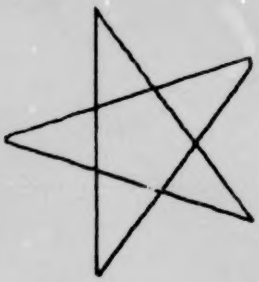


Figure 1.2

In a later class Albert decided to have the computer draw his initials. His problems with this project illustrate his difficulty in understanding the state of the TURTLE and the use of subprocedures. He was encouraged to make an A and a J as separate figures and then write a subprocedure putting the two initials together.

In making his A, Albert estimated by eye, getting a close approximation to a symmetrical A. He first turned the TURTLE 20 degrees to the right, and made the first leg of the A. He then used 144 degrees (borrowed from his star) for the vertex angle. (A rotation of 140 degrees would have made an exactly symmetrical A. The rest of the design remained skewed at 4 degrees from the vertical and horizontal axes.)

Although making the subprocedure J posed no great problem for Albert, putting the A and the J together proved to be more difficult. Albert usually worked on a Turtle Geometry project by trying out all the steps first, and then copying them over to make a procedure. He used this approach for making both of the letters for his initials. When he came to making a procedure to put them together, however, he did not seem to realize that he could work in the same step-by-step manner, using the subprocedures, A and J as though they were direct commands. Instead he wrote the procedure first, then tested and debugged it.

In addition to not realizing that he could use the A and J as though they were direct commands, Albert had difficulty realizing how the state of the TURTLE effected the position of the J. Although he had an excellent grasp of the state of the TURTLE as it related to forward, back right and left commands, he seemed to lose track of it as it related to a procedure -- a combined sequence of commands. This was similar to the problem he had in predicting where the RCIRCLE and

LCIRCLE commands would draw certain features of his face in an earlier project.

An analysis of Albert's separate procedures to draw an A and a J is shown below. The TURTLE starts drawing the A at point 1. It finishes the A at point 2. It starts and finishes the J at points 3 and 4 respectively (see figure 1.3). In his work however, Albert seemed unable to take these TURTLE states into account.

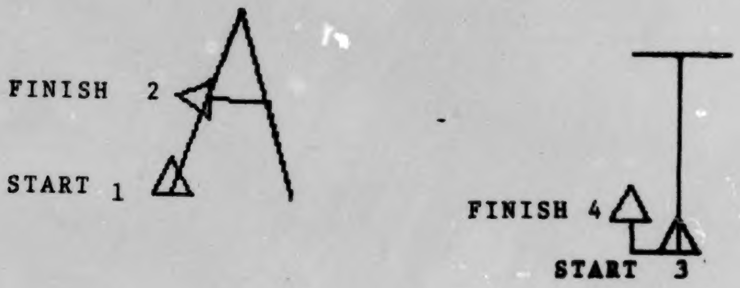


Figure 1.3

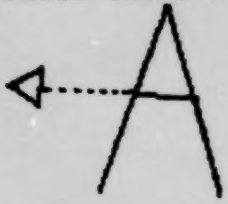
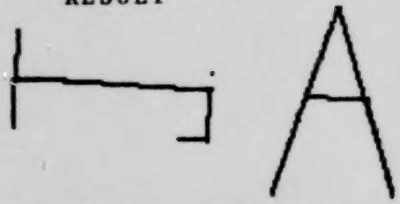
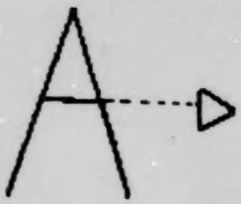
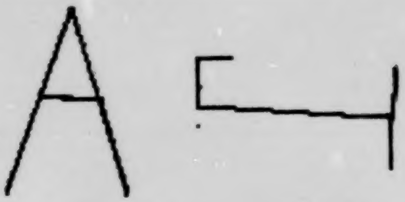
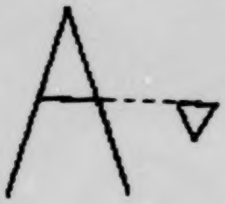
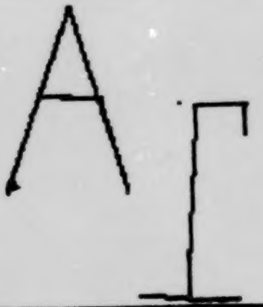
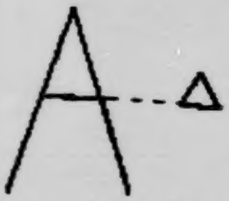
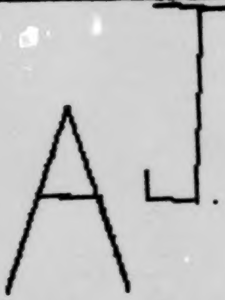
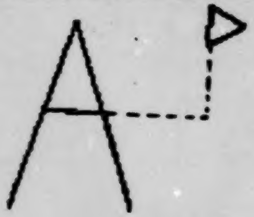
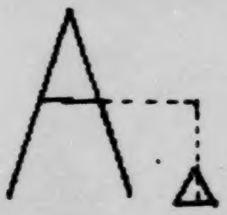
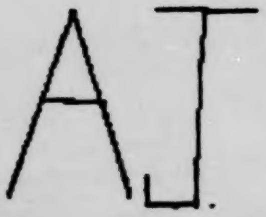
It took Albert seven tries to achieve his procedure MAME (a substitute for the desired title "NAME", an existing LOGO command). Although each try got closer to the desired result, the whole approach seemed devoid of planning. Figure 1.4, on the next page, shows the sequence of Albert's trials. The dotted line on the left shows where the turtle moved before making the J.

1.2 Albert's Use of Procedures and Subprocedures

Albert quickly understood basic TURTLE commands, and the idea of combining a string of TURTLE commands to write a procedure. As we have seen, he developed a strategy of trying out a sequence of direct commands, and, if he liked the design, copying the entire list of commands to make a procedure. He easily learned to edit procedures to correct spelling or syntax errors and became proficient with filing. On the other hand, Albert had difficulty debugging procedures in which there were errors of substance rather than syntax. His programming work, often suffered from a series of related confusions which will be described in detail below.

> --he failed to use step-by-step analysis of his procedures as a debugging tool;
>
> --he had difficulty understanding the role of subprocedures as separate entities;

Figure 1.4

TRIAL	TURTLE MOVED TO	RESULT
1		
2		
3		
4		
5	SAME AS TRIAL #3	SAME AS TRIAL #3
6		THE J WAS "OUT OF BOUNDS"
7		

--he used both recursion and variables in a "mechanical" fashion, without really understanding how they worked.

In the third class, Albert used the TURTLE to draw a rather involved "face". He copied down more than 35 steps in writing a procedure called KEITH. (See Figure 1.1). When copying errors -- the omission of two or three necessary steps -- led to bugs, Albert was unable to resolve the problem without help. He was shown how to use the STEP command, and was able to successfully edit his procedure using it. However, in later work, he rarely applied this approach on his own -- preferring to start over completely, rather than go through a step-by-step debugging of a previously defined procedure.

While Albert learned how to use previously defined procedures as building blocks, he had great difficulty using the idea of subprocedures in his planning. He seemed to have a fixed idea of a procedure as a sequential list of commands. When a procedure was used as a subprocedure, he seemed to lose track of how it fit into his overall purpose. His "initials" project, serves as a good example of this.

In another project -- making a "spacewar" animation in which a "UFO" orbited around a "planet", while shooting at it, Albert used subprocedures at the teacher's suggestion. When bugs occured, he had difficulty realizing which subprocedure as at fault. He put subprocedures in the wrong places, often using them more than once. He needed a good deal of help to debug his procedures successfully.

It was not until the last few classes that he began to incorporate subprocedures into his work independently.

A similar situation occured in Albert's work with recursion. He easily learned to use recursion in a "cliche" form, but had difficulty debugging improper uses of recursion. During one class, for example, Albert created a number of recursive designs. One of them, FLAIT, was intended to produce an effect something like an asterisk. First he defined one "ray" of the asterisk:

```
TO FLAIT
1 FORWARD 56
2 RIGHT 90
3 FORWARD 2
4 RIGHT 90
5 FORWARD 56
END
```

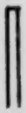


Figure 1.5

What he needed to do next was to rotate the TURTLE, and use recursion to repeat the whole process. What he did was to add the recursion line without including the necessary rotation:

```
TO FLAIT
1 FORWARD 56
2 RIGHT 90
3 FORWARD 2
4 RIGHT 90
5 FORWARD 56 FLAIT
END
```

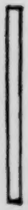


Figure 1.6

When this version of FLAIT did not produce the desired effect, Albert tried to use it as a subprocedure of 9FLAIT:

```
TO 9FLAIT
1 FLAIT
2 FLAIT
3 FLAIT
4 FLAIT
5 FLAIT
6 FLAIT
7 FLAIT
8 FLAIT
9 FLAIT
END
```

He was surprised that 9FLAIT produced exactly the same effect as FLAIT, and gave up on his project. Several weeks later, however, when asked why 9FLAIT and FLAIT had the same effects, Albert responded quite easily: "Because FLAIT never stops," and indicated that only line 1 of 9FLAIT was ever executed. He never seemed to understand his original bug -- forgetting to put a rotation before the recursion line in FLAIT.

Albert also learned to use variables to change the angle or size of a given predetermined shape. Again, he was able to debug problems that arose due to syntax errors, but had difficulty with those that had to do with a misconception of the purpose of the variable or the way in which it was assigned.

For example, Albert taught the computer to draw a 5-pointed star (Figure 1.2).

```
TO STAR
1 FORWARD 100
2 RIGHT 144
3 FORWARD 100
4 RIGHT 144
5 FORWARD 100
6 RIGHT 144
7 FORWARD 100
8 RIGHT 144
9 FORWARD 100
10 RIGHT 144
END
```

Later, during class 10, he was shown how to make a star of variable size by substituting :SIZE for the value 100 in all the forward steps, and using :SIZE in the procedure title. His new procedure was:

```
TO AS :SIZE
1 FORWARD :SIZE
2 RIGHT 144
3 FORWARD :SIZE
4 RIGHT 144
etc.
```

Later, however, Albert had difficulty applying this idea to the task of designing a racetrack. He was drawing a shape consisting of two boxes, one inside the other, and wanted to write a procedure that could draw boxes of different sizes. He had begun his work by moving the TURTLE from the origin to the lower left hand corner of the screen, and drawing a large box, using these commands:

```
PENUP LEFT 90
FORWARD 200
RIGHT 90
BACK 200
```

These steps move the TURTLE over to the lower left hand corner of the screen.

```
PENDOWN FORWARD 375
RIGHT 90
FORWARD 375
RIGHT 90
FORWARD 375
RIGHT 90 FORWARD 375
RIGHT 90
```

These steps draw a box with sides of 375.

When it was suggested to Albert that he make a variable box procedure, he followed the same process he did with his STAR: he copied the steps, substituting :SIZE for the originally fixed length in all the FORWARD steps without changing the RIGHT or BACK step. He had not differentiated the steps needed to move the TURTLE over, from those needed to draw the box:

```
TO BX :SIZE
5 PENUP LEFT 90
10 FORWARD :SIZE
15 RIGHT 90
20 BACK 200
25 PENDOWN FORWARD :SIZE
30 RIGHT 90
35 FORWARD :SIZE
40 RIGHT 90
45 FORWARD :SIZE
50 RIGHT 90 FORWARD :SIZE
55 RIGHT 90
END
```

When this produced an off-center box, Albert was mystified. He debugged this procedure in step by step fashion, first removing LEFT 90 from line 5 and then erasing line 20. Although his BX procedure now "worked", he never understood his original error. We can see that he was still confused <u>both</u> about the use of variables to replace fixed steps in variablizing a procedure, <u>and</u> about separating his problem into subparts that had different functions.

Albert's problems were compounded by the fact that he asked for help only when totally frustrated. By that point he had usually tried so many alternatives that he was no longer aware of what his original difficulty had been. In a class in which the other three students (Dennis, Harriet and Tina) were extremely demanding, Albert, working quietly on his own, tended to get lost. The problems he encountered remained confusions, and were rarely used as vehicles to help him improve his understanding, planning and organization.

1.3 Albert's Work on Animation Projects.

Work involving animation of different kinds was a major theme of Albert's LOGO experience. He began incorporating SPINs in his designs in the second and third class. In the seventh and eighth classes he developed a "Starwars" project that involved using positive and negative SPINs to draw a UFO orbiting around a planet. This project also made use of display commands, SNAP, DISPLAY and RUBDIS to create a shooting effect, and to make the planet disappear. In a later class he drew a car, and animated it using SPIN and MOVET. He then got very absorbed in the process of designing tracks for the car to race on. Still later, Albert animated a gocart using MOVET. He had a major problem orienting the TURTLE so that the car was drawn horizontally and moving horizontally from left to right.

In class nineteen Albert was given procedures to animate the TURTLE, and shown how to write a procedure that would allow him to change the TURTLE's motion by typing letters on the keyboard. He was given the procedure DT, and the idea for the procedure CHANGE. (KEY was given to him as a "primitive" which " sends a message to tell which key on the keyboard you typed.") The procedures were:

```
TO DT                TO CHANGE
10 MAKE "D 10        10 MAKE "LETTER KEY
20 PENUP             20 IF :LETTER = "R RIGHT 20
30 FORWARD :D        30 IF :LETTER = "L LEFT 20
40 CHANGE            40 IF :LETTER = "F MAKE "D :D+10
50 GO 30             50 IF :LETTER = "B MAKE "D :D-10
END                  60 IF :LETTER = "C RCIRCLE 20
                     END
```

Albert himself chose the letters to use in CHANGE and decided what changes would occur when those letters were typed.

Now Albert settled into an area that he could understand -- designing racetracks for the moving TURTLE. He designed several: one was box shaped, another oval, a third an oval with a figure eight in it. Throughout Albert's work with animation it was clear that he was not especially concerned with how the animation occurred, but rather with using the animated object in a context of his own design. Therefore the last activity, where Albert was given procedures with which he could carry out a variety of projects, was the most successful. Although he could explain how the procedures DT and CHANGE worked, he could not derive similar ones himself. He did change the effect, by editing CHANGE several times to alter the angle turned, or the amount the distance increased and decreased and to add a new procedure which made the TURTLE jump instantaneously across the screen.

2. Betsy

Betsy is a slow, methodical student who worked on several long projects, each lasting for a number of class periods. Although she was quiet, and rarely asked for help in her work, she responded well to suggestions from her teacher and once having acquired a new idea she was usually able to make use of it in different contexts. Betsy is considered to be below average in overall academic ability. Her most recent achievment test scores place her in 32nd percentile, when compared to national norms. Our presentation of Betsy's work focuses on examples of her major projects.

1. Betsy's Horse Project

Betsy's first project was to draw a horse, using the computer. In the course of working on this project, Betsy learned to estimate angles and distances with the TURTLE, to make use of 90 degree angles; to draw rectangular shapes; and to make use of right/left reversibility. In the area of programming she learned to write procedures to use subprocedures to draw different parts of her horse, and to write a superprocedure to put the entire project together.

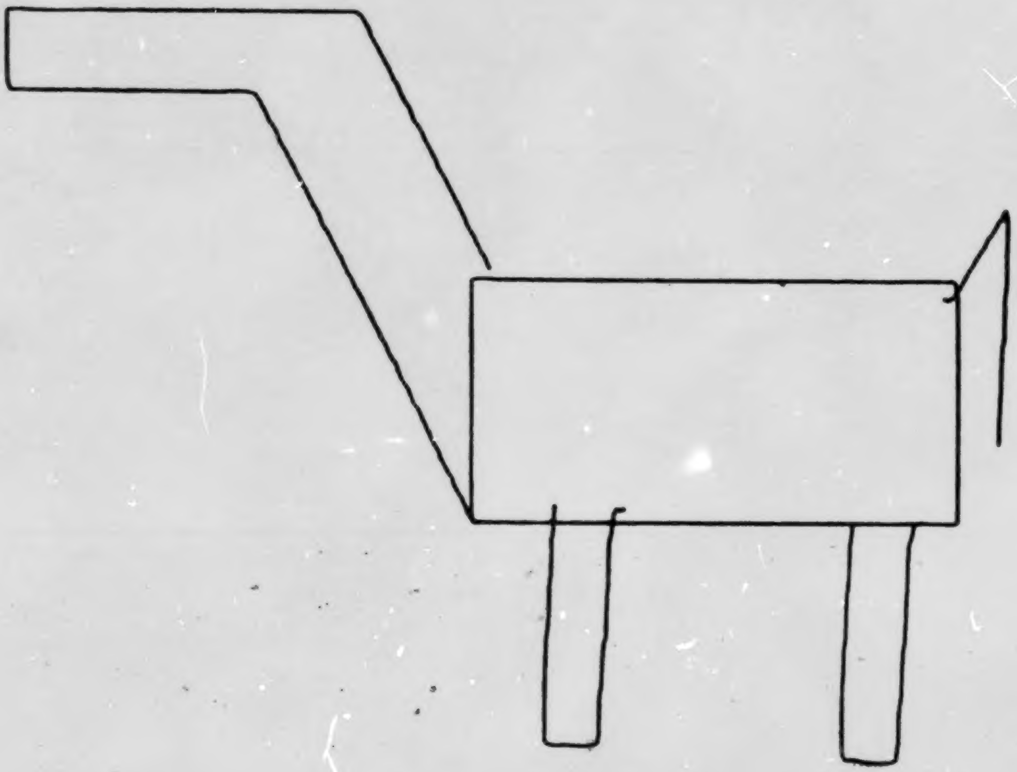


Figure 2.1

```
TO HORSE
1 BODY
2 HEAD
3 TAIL
4 LEG
5 SEE
END
```

HORSE is a superprocedure which draws the entire horse

```
TO BODY
10 FORWARD 90
20 RIGHT 90
30 FORWARD 90
40 FORWARD 90
50 RIGHT 90
60 FORWARD 90
70 RIGHT 90
80 FORWARD 90
90 FORWARD 90
END
```

BODY draws a rectangle to represent the horse's body

```
TO HEAD
10 RIGHT 63
20 FORWARD 180
30 LEFT 63
40 FORWARD 90
50 RIGHT 90
60 FORWARD 20
70 FORWARD 10
80 RIGHT 90
90 FORWARD 90
100 FORWARD 40
110 RIGHT 63
120 FORWARD 90
130 FORWARD 20
140 LEFT 60
END
```

HEAD draws the head and neck of the horse

```
TO TAIL
10 PENUP FORWARD 170
20 LEFT 63
30 PENDOWN
40 FORWARD 40
50 RIGHT 63
60 RIGHT 90
70 FORWARD 90
END
```

TAIL moves the TURTLE over from where the neck finishes, and draws the horse's tail

```
TO LEG
10 RIGHT 90
20 PENUP
30 FORWARD 20
40 LEFT 90
50 FORWARD 30
60 RIGHT 90
70 FORWARD 10
80 LEFT 90
90 PENDOWN
100 FORWARD 90
110 RIGHT 90
120 FORWARD 20
130 RIGHT 90
140 FORWARD 90
END
```

LEG moves the TURTLE from where the tail finishes, and draws the horse's rear leg

```
TO SEE
10 LEFT 90
20 PENUP
30 FORWARD 90
40 LEFT 90
50 PENDOWN
60 FORWARD 90
70 RIGHT 90
80 FORWARD 20
90 RIGHT 90
100 FORWARD 90
END
```

SEE moves the TURTLE over to draw the front leg

Having already called one procedure LEG, Betsy chose a miscellaneous name, SEE, for the second leg. Since she used the same subprocedure to move the TURTLE over, and to draw each feature, the subprocedure LEG, could not be used again to draw the second leg.

2.2 Geometric Designs Using Arcs

Betsy spent six class periods making a series of designs with quarter arcs. She enjoyed the effect made by alternating left and right arcs. Her procedure, BOX, drew the largest series of four alternating arcs that would fit on the screen without producing an "OUT OF BOUNDS" message:

```
TO BOX
10 RARC 49
20 LARC 49
30 RARC 49
40 LARC 49
END
```

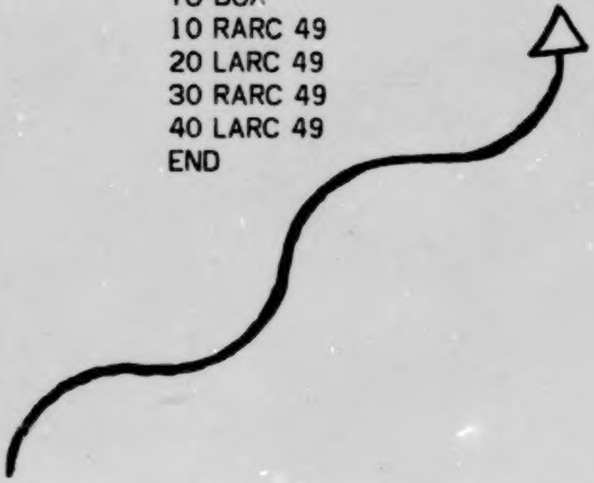


Figure 2.2

```
TO JIMMY
10 BOX
20 LEFT 90
30 LEFT 90
40 BOX
END
```

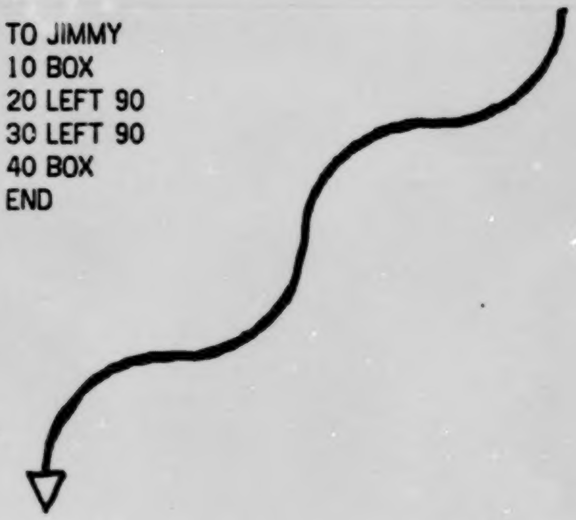


FIGURE 2.3

Her procedure, JIMMY, made the TURTLE retrace its steps so that it returned to its point of origin. When she repeated the same process four times, to make a symmetrical design, KATHY, she gave each of her four subprocedures a different name. At this point she followed the same pattern of subproceduralization that she had in her horse project: each part of the design had its own function, and hence its own name. While she was consciously repeating the same <u>process</u> four times to make the overall design, she was not yet comfortable with the idea of repeating the same <u>procedure</u> four times -- although that had been suggested by the teacher.

```
TO KATHY
10 JIMMY
20 KARL
30 SUSAN
40 LAURA
END
```

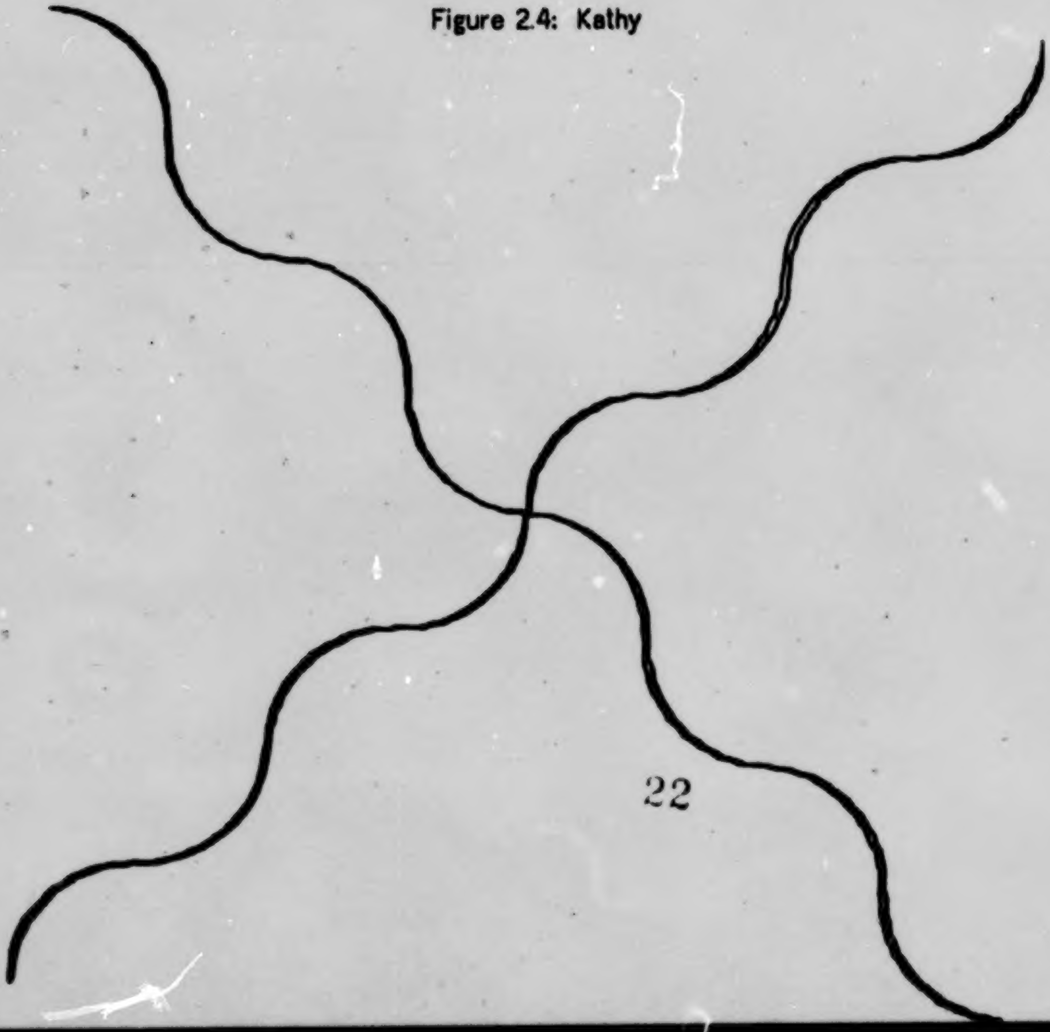
```
TO JIMMY
10 BOX
20 LEFT 90
30 LEFT 90
40 BOX
END
```

```
TO KARL
10 BOX
20 LEFT 90
30 LEFT 90
40 BOX
END
```

```
TO SUSAN
10 RIGHT 90
20 BOX
30 LEFT 90
40 LEFT 90
50 BOX
END
```

```
TO LAURA
10 BOX
20 LEFT 90
30 LEFT 90
40 BOX
END
```

Figure 2.4: Kathy



Betsy's next design was made after a period of experimentation with repeating BOX, rotation, BOX, rotation, ... This time she made a subprocedure, RAY, consisting of BOX and a rotation, and found that repeating RAY nine times produced a "sun".

```
TO RAY
10 BOX
20 LEFT 180
30 RIGHT 20
END
```

```
TO SUN
10 LEFT 130
20 PENUP
30 FORWARD 180
40 RIGHT 130
50 PENDOWN
60 REPEAT [RAY] 9
END
```

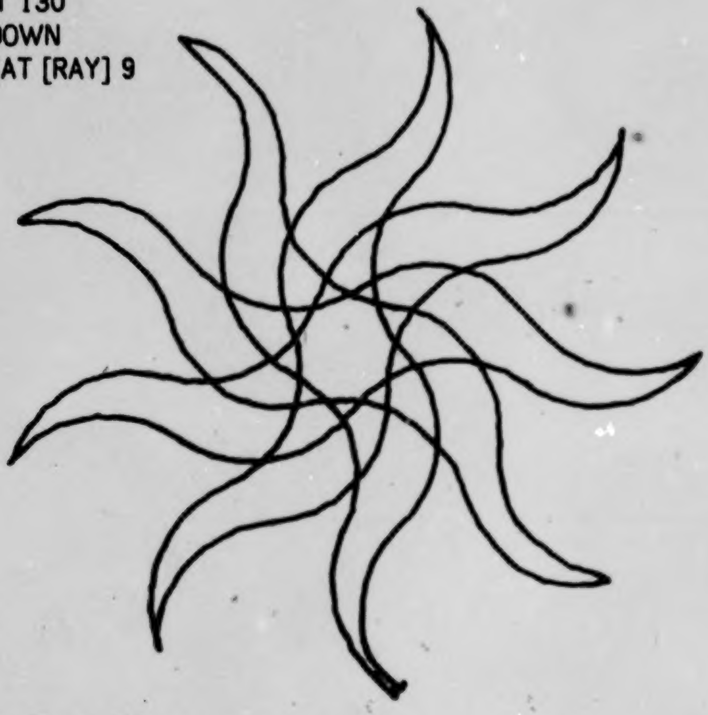


Figure 2.5

Lines 10-50 of SUN are needed to move the TURTLE over so that the entire design appears on the screen. The design is drawn by line 60: REPEAT [RAY] 9.

Next, Betsy wanted to make suns of different sizes. She learned the syntax for using variables, and rewrote her procedures:

```
TO BOX2 :SIZE
10 RARC :SIZE
20 LARC :SIZE
30 RARC :SIZE
40 LARC :SIZE
END
```

```
TO RAY2 :SIZE
10 BOX2 :SIZE
20 LEFT 180
30 RIGHT 20
END
```

```
TO SUN2 :SIZE
10 REPEAT [RAY2 :SIZE] 9
END
```

Having defined a variable sun, Betsy set out to make a "sky" full of stars. Her design was limited to two stars, by the memory limitations of the display computer. Betsy's superprocedure, BUD, which draws the sky, uses the subprocedure SUN2, twice.

```
TO BUD
10 PENUP
20 LEFT 90
30 FORWARD 180
40 RIGHT 180
50 PENDOWN
60 FORWARD 370
70 LEFT 120
80 PENUP
90 FORWARD 90
100 PENDOWN
110 SUN2 10
120 PENUP
130 BACK 20
140 LEFT 120
150 RIGHT 45
160 RIGHT 10
170 FORWARD 180
180 RIGHT 90
190 FORWARD 70
200 PENDOWN
210 SUN2 10
END
```

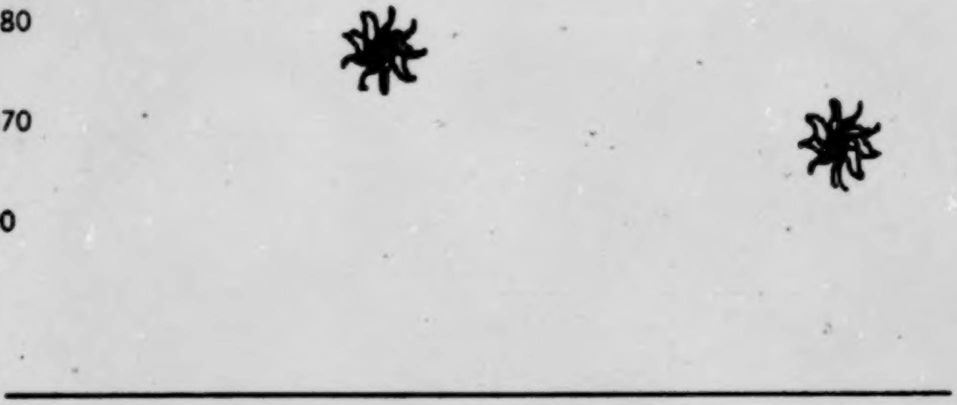


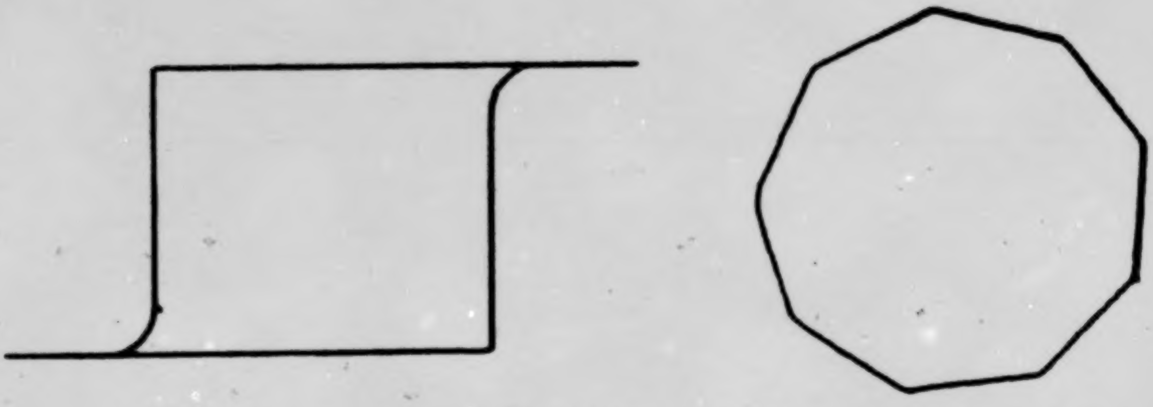
Figure 2.6

2.3 Recursive Projects

For a few periods, Betsy experimented with randomly creating geometric designs by using recursion. ABC and SWING were two designs which she liked well enough to make printed copies:

```
TO ABC
10 FORWARD 90
20 RIGHT 90
30 FORWARD 180
50 LEFT 180
60 FORWARD 40
70 LARC 20
80 ABC
END
```

```
TO SWING
10 FORWARD 50
20 LEFT 40
30 SWING
END
```


ABC

SWING

Figure 2.7

This led rather naturally to making a series of designs using POLY, and exploring the relation betweeen the angle inputs to POLY, and the resulting shape. These explorations, and designs made using POLY occupied another few classes.

2.4 Debugging a Person

Near the end of the series of classes, Betsy was asked to debug a set of deliberately buggy procedures, designed to draw a person. She was faced with the problem of debugging a set of subprocedures which drew a figure like this:

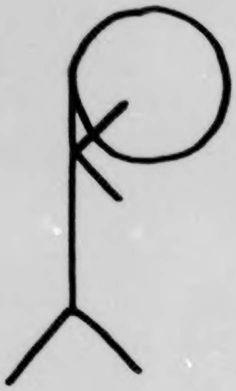


Figure 2.8

Betsy had no difficulty getting the arms oriented correctly. She did have difficulty positioning the TURTLE to draw the head, however. Rather than make use of the standard "fix" of rotating the TURTLE 90 degrees before drawing the head. Betsy used a more elaborate process of moving the TURTLE upward and to the left before starting to draw the head.

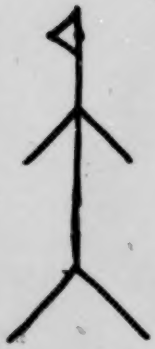


Figure 2.9a

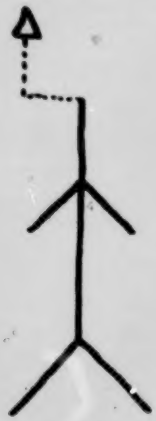


Figure 2.9b

Figure 2.9

The resulting figure, and the procedures which were used to draw it are shown below.

```
TO PERSON
10 BOD
20 LEGS
30 ARMS
40 HEAV
END

TO BOD
10 FORWARD 20
20 BACK 80
END

TO LEGS
10 RIGHT 180
20 LEFT 45
30 FORWARD 40
40 BACK 40
50 RIGHT 90
60 FORWARD 40
70 BACK 40
80 LEFT 45
END
```

```
TO ARMS
10 RIGHT 180
20 FORWARD 60
30 RIGHT 90
40 RIGHT 45
50 FORWARD 30
60 BACK 30
70 RIGHT 90
80 FORWARD 30
90 FORWARD 30
100 RIGHT 90
110 RIGHT 90
120 LEFT 45
END
```

```
TO HEAV
10 FORWARD 30
20 PENUP
30 LEFT 90
40 FORWARD 20
50 RIGHT 90
60 FORWARD 25
70 PENDOWN
80 RCIRCLE 30
END
```

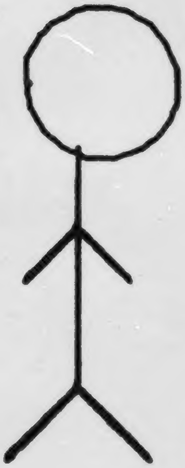


Figure 2.10

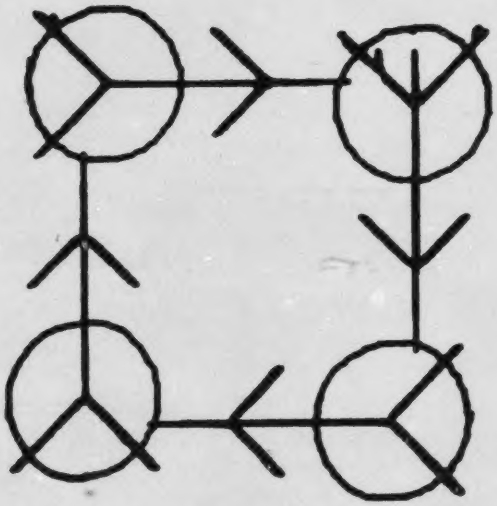
Betsy's work debugging person was followed by a recursive design, which produced a string of overlapping figures,

```
TO PEOPLE
10 PERSON
20 PEOPLE
END
```

and a design drawn by arranging four PERSONs in a kind of square:

```
TO PERSO
10 PERSON
20 RIGHT 90
30 PENUP
40 FORWARD 90
50 PENDOWN
60 PERSON
70 RIGHT 90
80 PENUP
90 FORWARD 90
100 PENDOWN
110 PERSON
120 RIGHT 90
130 PENUP
140 FORWARD 90
150 PENDOWN
160 PERSON
END
```

Figure 2.11



3. Darlene

Darlene is a bright, quietly attractive student, who is considered to have "above average" ability by her teachers. Her most recent achievement test scores place her in the 49th percentile as compared to national norms. Despite the potential perceived by her teachers, she does not seem to challenge herself in school, or to respond to challenges from teachers. As one of her teachers put it: "Darlene always seems to exert herself to the extent necessary to get a B, and no more. She does not push herself, and is not concerned with doing the best she can." Darlene's school work is conscientiously neat and punctual, but not inspired.

Darlene exhibited much the same pattern in LOGO classes. Her work was characterized by a large number of short projects, usually involving attractive geometric designs. She very quickly learned a few key ideas in LOGO programming and Turtle Geometry, and applied these ideas over and over again in similar projects. She tended to reject suggestions that led to longer, more involved projects, or to more complex learnings.

Darlene had a lot of curiosity about the LOGO language, its commands and error messages. She often carried out "experiments" to test the limits of the language, or the computer system. She enjoyed examining the work of other children, which she was able to understand by printing out their procedures and stepping through them. She also became adept at the use of the LOGO filing system, and had several files of her own, which she learned to manipulate expertly.

Darlene learned to use all of the system peripherals: the floor turtle (during the few days it was available), the printer, and especially the plotter. She devoted a fair amount of effort to making plotter pictures of her designs, and using the plotter and printer output to make neat, well organized displays of her work.

Darlene had a desire to be self-sufficient in her work. She usually preferred to scrap a project with which she was having trouble, rather than to debug it carefully or ask for help. While she liked to learn new ideas that would have immediate effects (such as the use of REPEAT or SPIN comands) she rejected new ideas when they required a lot of concentration, time to implement, or a difficult new syntax (such as the use of variables).

3.1 Darlene's Explorations with the LOGO Language and the Computer System

Darlene exhibited a great deal of curiosity about the workings of the LOGO language and the computer system. This sometimes took the form of trying to get the computer to generate error messages, or to "printout" LOGO primitives. She was curious about the error messages generated by using alphanumeric and non-alphanumeric characters. When she received an error message she often responded by carring out a series of experiments. For example, the following sequence; taken from Darlene's dribble file:

```
?65
YOU DON'T SAY WHAT TO DO WITH 65
?H65
YOU HAVEN'T TOLD ME HOW TO H65
?6 + 2
YOU DON'T SAY WHAT TO DO WITH 8
?6 + 2=
=NEEDS MORE INPUTS
?6 + 2 = 8
YOU DON'T SAY WHAT TO DO WITH "TRUE
?PR 6 + 2 = 8
8
=NEEDS MORE INPUTS
```

In the same session, she generated the longest single "word" that the computer would accept, typing random characters for twenty-one lines, until the computer printed a "NO MORE STORAGE" error message.

In another session Darlene explored a rather confusing series of errors. Although she did not ask for help, and the teacher was not directly aware of what she was trying to do at the time, we have been able to reconstruct her mode of operation by a detailed analysis of her dribble file:

?TO &&**	Darlene is experimenting with non-alphanumeric procedure names
* CAN'T BE AN INPUT	
	Without understanding the message, she picks a new name, again non-alphanumeric
?TO &)&)	
>10 FORWARD 12 >20 RIGHT 90 >30 FORWARD 12 >40 RIGHT 90 >50 FORWARD 12 >60 SPIN 100 >END	Getting no error message, Darlene defines a procedure, using a familiar sequence of commands
& DEFINED	Darlene noticed that the computer responded

"& DEFINED" and used & as her procedure title, rather than &)&)

```
?&
& NEEDS MORE INPUTS
```

Noticing the error message, she now tries her original title, &)&)

```
?&)&)
UNEXPECTED RIGHT PARENTHESES
?ED &
```

Darlene realizes that something about her procedure needs to change, but she is not sure what to do, so types END, and tries some more experiments

```
>END
& DEFINED
?&
& NEEDS MORE INPUTS
```

THis time she tries an "input"

```
?& '(
YOU HAVEN'T TOLD ME HOW TO '
?&78
```

Tries a numerical input, but forgets to leave a space

```
YOU HAVEN'T TOLD ME HOW TO &78
?& 56
& NEEDS MORE INPUTS
?POTS
TO & ) & )
```

Here Darlene gets another "clue" about how the computer "thinks" &)&) should be typed

```
?TO & ) & )
```

She copies the computers version, including TO

```
YOU HAVE ALREADY TOLD ME HOW TO &
?ER & ) & )
```

Using the computer's version again

```
UNEXPECTED RIGHT PARENTHESES
?POTS
TO POLY :SIDE :ANGLE
```

Darlene notices that despite the error message, &) &) has been erased.

```
?ER ALL
```

She clears the workspace and starts again

```
? READ "DEANNE >
?POTS
?TO POLY :SIDE :ANGLE
?TO &)&)
```

She tries again!

```
>10 FORWARD 12
>20 RIGHT 90
>30 FORWARD 12
>40 RIGHT 90
```

```
>50 FORWARD 12
>60 SPIN 100
>END
& DEFINED
POTS
TO & ) & )
TO POLY :SIDE :ANGLE
```

Darlene doesn't even try & this time.

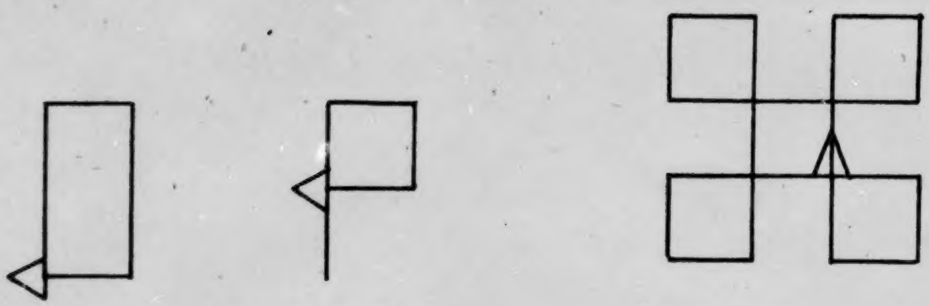
When she sees that the computer has printed it out the same way, she knows it won't work, and just erases it. This time she erases it using <u>her</u> version of the title, and checks, using POTS, to see if the procedure was erased.

```
?ER &)&)
UNEXPECTED RIGHT PARENTHESES
?POTS
TO POLY :SIDE :ANGLE
?TO TOO
```

During this entire sequence, Darlene never asked for help, or mentioned her difficulty. Finally abandoning the struggle, she chose a more conventional name for her procedure, TOO, and went back to her original plan. Her difficulties stemmed from two problems. Not understanding the special role of parentheses in LOGO, and being relatively unfamiliar with variables. One experiment she did not try was to continue adding inputs, typing <u>&</u> 58 58 58 for example, until the error message, "& NEEDS MORE INPUTS" no longer occurred. Perhaps a greater familiarity with multi-variable procedures might have led her to that experiment. At a minimum, she probably learned that non-alphanumeric characters can cause difficulties in procedure titles. Her persistence, in exploring the system on her own, without help, was characteristic of her work.

2. <u>Darlene's Use of Procedures</u>

Darlene had an excellent sense of a procedure as an entity. She began defining and repeating procedures in the first class. When a procedure that was supposed to make a "D", (see Figure 3.1a): had a bug in it (see Figure 3.1b), she immediately repeated the procedure four times, until the shape "closed" (Figure 3.1c).

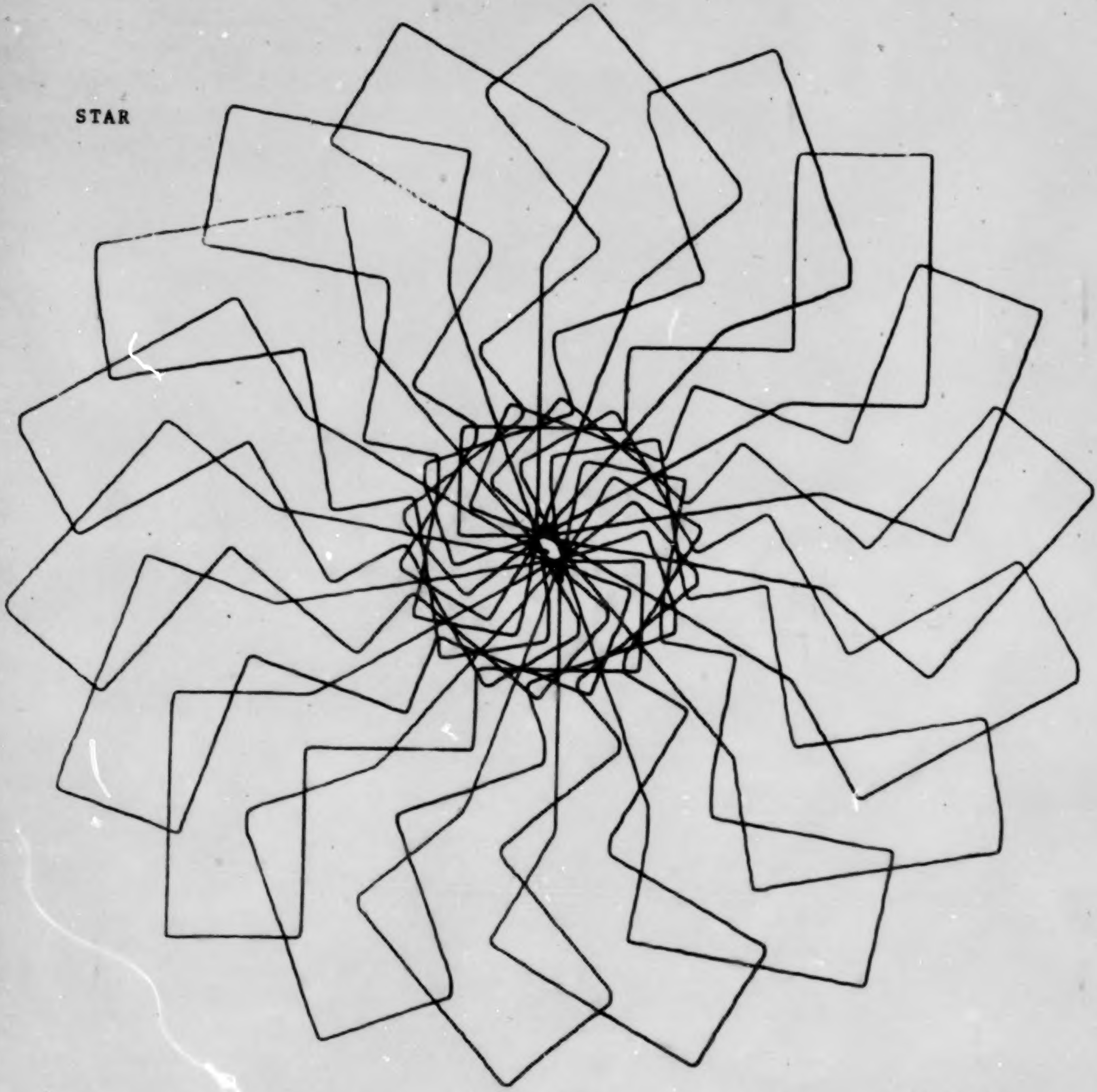


3.1a 3.1b 3.1c

Figure 3.1

Making patterned procedures remained a major theme of Darlene's work. Darlene was also able to establish a clear hierarchy in the use of subprocedures. In the fourth class, she made an experimental design which she called STAIRS. Having finished the design, she began repeating it. After a few repeats, she saw that it was going to require a large number of repeats before it "closed". She now began to use the REPEAT command, finally determining that her figure closed after 18 repeats of STAIRS. She made this a procedure, and also defined a procedure which would cause the whole shape to spin:

STAR



```
TO STAR
10 REPEAT [STAIRS]18
END
```

```
TO STARSPIN
10 SPIN 100
20 STAR
END
```

STAIRS

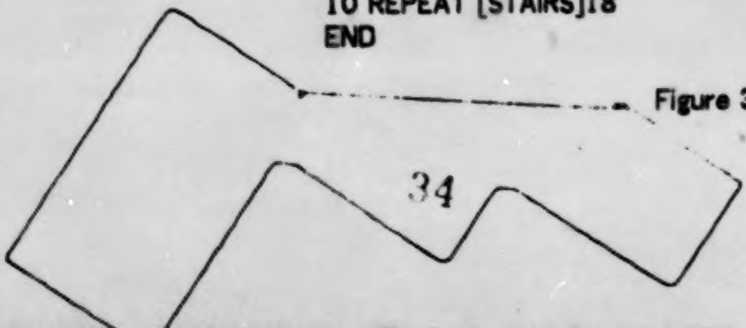


Figure 3.2

This approach was a common one in Darlene's work: make some kind of original design, then repeat it until it makes a closed or complete figure. When Darlene made a design which included a SPIN command, she quickly transformed it into a repeated spin with her second procedure.

```
TO POINT
10 FORWARD 70
20 RIGHT 67
30 FORWARD 2
40 RIGHT 67
50 FORWARD 56
60 SPIN 100
END
```

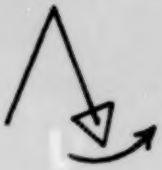


Figure 3.3

```
TO ZIPPER
10 REPEAT [POINT]10
END
```

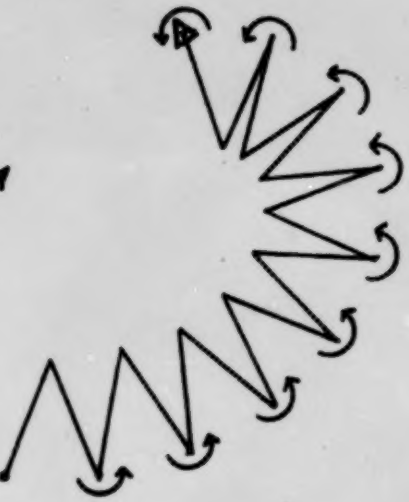


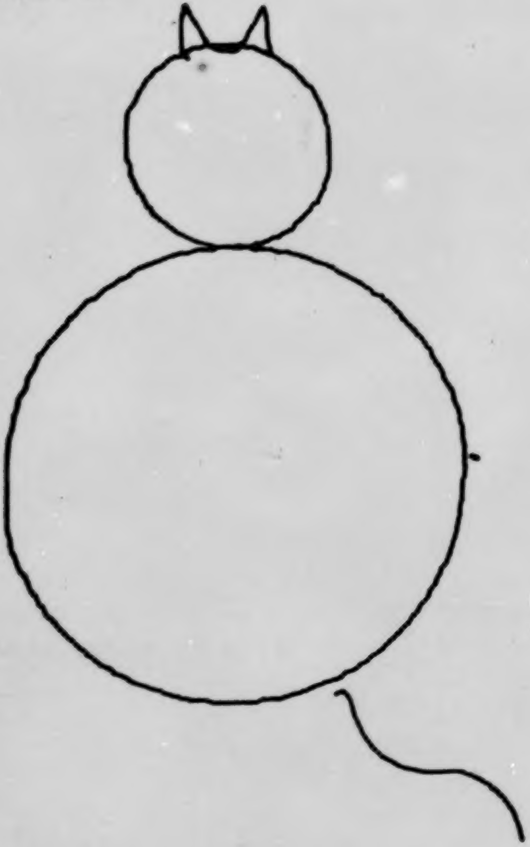
Figure 3.4

The name, ZIPPER, came from the visual effect cause by the shape of the module and the multiple spins.

In addition to having a clear sense of the hierarchical structure of LOGO procedures, Darlene also understood how to break a long project into shorter subprocedures, in order to simplify the construction process. Her procedure CAT, was made up of six subprocedures, each of which had a clearly defined function:

```
TO CAT
10 RIBIT          ;draws two circles
20 WEE            ;moves the TURTLE to draw the first ear
30 EAR1           ;draws the first ear
40 EAR2           ;moves the TURTLE and draws the second ear
50 TURN           ;moves the TURTLE to draw the tail
60 TAIL           ;draws the tail
70 HIDETURTLE
END
```

Figure 3.5



At some point after finishing the CAT project, Darlene decided to carry out a complex sequential Turtle Geometry project without using subprocedures. Her resulting procedure, CASTLE, included 63 steps (numbered by fives) and was accomplished with a minimum of debugging.

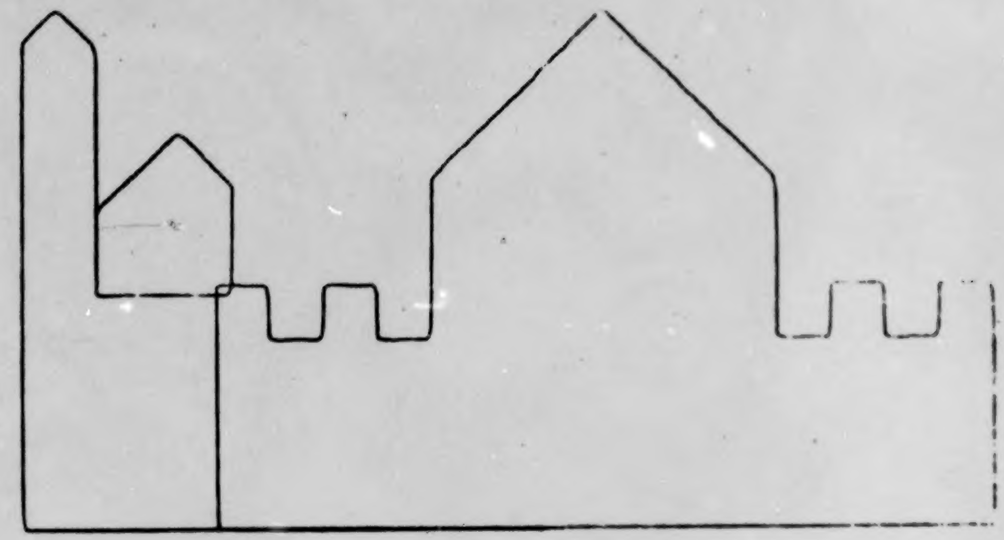


Figure 3.6

Darlene also enjoyed examining the work of other children. She would LOGIN with other children's names, read their files, and try out their procedures. She would expertly trace her way through a procedure hierarchy to find out what each subprocedure did and how it worked. Once she copied a set of procedures from her classmate, Betsy, expertly changing each subprocedure name, so that Darlene's procedures produced the same effect as Betsy's with different names:

Betsy's Procedures

```
TO BOX
10 RARC 49
20 LARC 49
30 RARC 49
40 LARC 49
END

TO RAY
10 BOX
20 LEFT 180
30 RIGHT 20
END

TO SUN
10 LEFT 130
20 PENUP
30 FORWARD 180
40 RIGHT 130
45 PENDOWN
50 REPEAT[RAY]9
END
```

Darlene's Procedures

```
TO PIG
10 RARC 49
20 LARC 49
30 RARC 49
40 LARC 49
END

TO GLASS
10 PIG
20 LEFT 180
30 RIGHT 20
END

TO SMILE
10 LEFT 130
20 PENUP
30 FORWARD 180
40 RIGHT 130
45 PENDOWN
50 REPEAT [GLASS]9
END
```

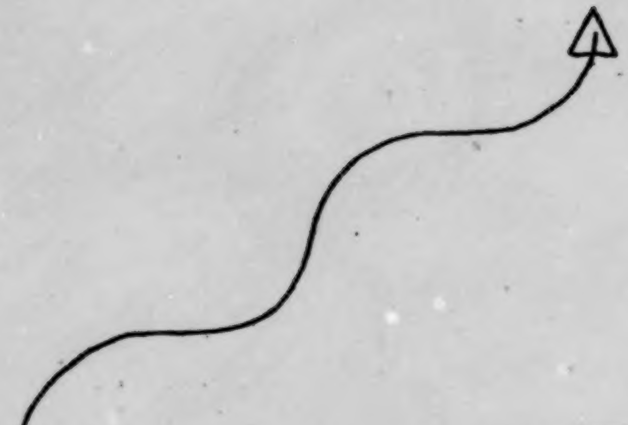


Figure 3.7a
PIG

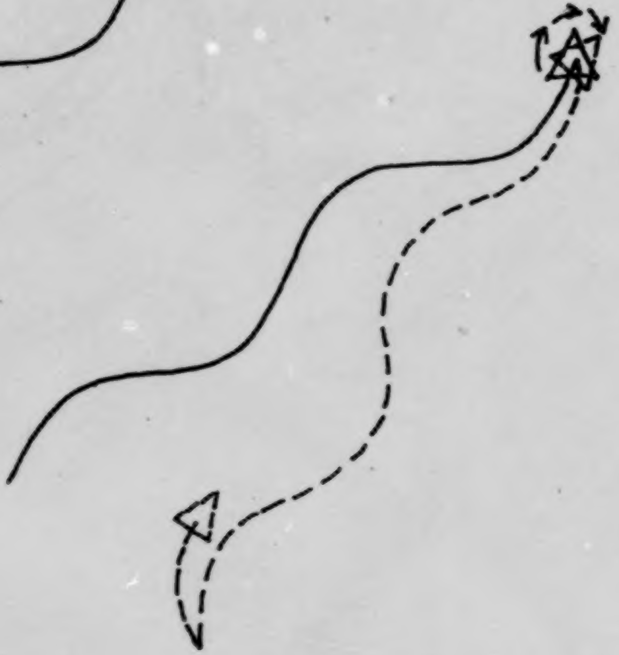


Figure 3.7b
GLASS

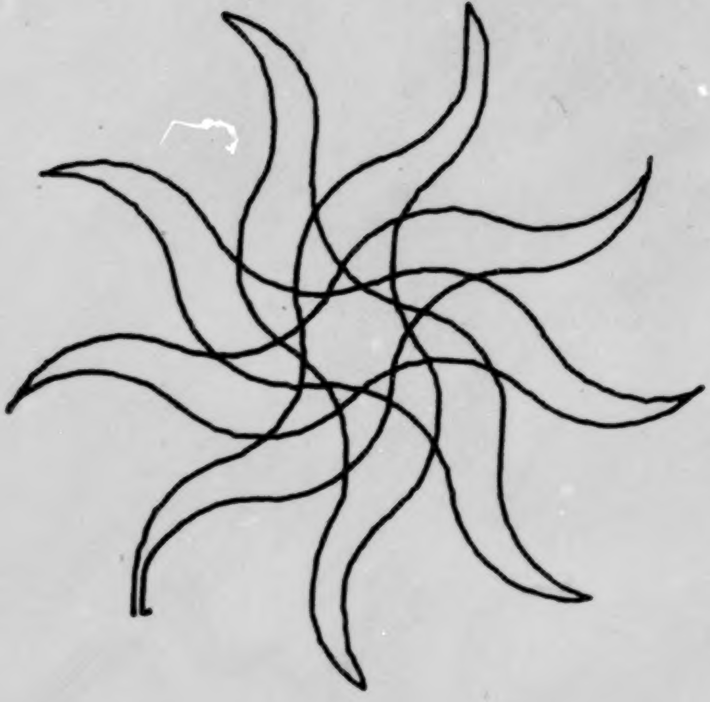


Figure 3.7c
SMILE

Although Darlene was expert at many aspects of LOGO she was confused by variables, and although she carried out a number of projects involving variables, she never made a consistent effort to understand either the syntax or the use of variables. Efforts to explain the use of variables, and project suggestions that would have allowed her to explore some simple uses of variables on her own, were met by Darlene with a great deal of resistance. It was not until the class began to work on dynamics activities during the last two weeks, that Darlene took a serious interest in variables. At this point she wanted to control the response of the DYNATURTLE to a KICK command. Darlene was given the procedure, COMMAND:

```
TO COMMAND
10 MAKE "LETTER KEY
20 IF :LETTER = "R RIGHT 30
30 IF :LETTER = "L LEFT 30
40 IF :LETTER = "K KICK 30
END
```

Darlene's problem was to make the dynaturtle reach a target. Many of her strategies resulted in the TURTLE missing the target and drifting off the screen (see Part II, Chapter 6, Dynamics, for more about Darlene's work with dynaturtle). To compensate for this Darlene decided that if the kick had a stronger effect the turtle would reach the target more easily. After being shown the COMMAND procedure, and experimenting with various inputs to KICK, she edited COMMAND to include an additional line:

```
50 IF :LETTER = "F KICK 2000
```

Later, she tried to make the TURTLE move more slowly so that it could land more softly on the target. This time she added the instruction:

```
60 IF :LETTER = "S KICK 10
```

In this way she began to understand the use and syntax of variables.

4. Deborah

Deborah is considered to be a below average student by her teachers. She has difficulty learning new topics, and is usually dependent on a lot of help from her teachers. She presents a good example of a "slow learner." Her most recent national achievement test scores place her in the 28th percentile.

Deborah began by being extremely timid and dependent in interactions with the computer. It was not until the eighth class that she had enough confidence to use the carriage return at the end of a line of instructions. She experienced great difficulty with simple projects. Starting at the 8th session, she was encouraged to "explore" with direct commands: FORWARD, BACK, RIGHT, LEFT, circles and arcs. She was able to gain confidence when exploring by limiting herself to very few commands and input numbers, which she repeated over and over. By intuitively choosing input numbers which make very nice designs (90s and 30s, for example), she was able to produce interesting effects. Gradually she learned to write procedures, to teach the computer to draw the designs she liked.

By the end of the series of classes she had created some unusual designs which won praise from her classmates, had carried out (with some help) a major project requiring the use of planning and subprocedures, and had a strong confidence in her ability to use the computer. She had invited both of her parents to visit the class, and they remarked to the LOGO teacher that this was the first time that Deborah had been excited about anything in school. Deborah's classroom teachers report that she has also become more assertive in class, has asked for extra help after school, etc.

1. Deborah's Working Style

The key to Deborah's success was her strategy of working with the computers. Deborah was able to slowly build her confidence and understanding by limiting her choices of LOGO commands and inputs, limiting the goals of her work, and by working in a way that minimized the chances of error. It was as if Deborah invented an unstated set of "rules" governing her work in LOGO which helped her to be successful.

> --she used a severely limited number of commands: FORWARD, RIGHT, LEFT, RARC, and LARC;
>
> --she limited the inputs she used with these commands to multiples of 10, up to 100. If a larger effect was needed, she would use additional steps, as in FORWARD 90, FORWARD 30. In fact, Deborah began by

> using only inputs of 30, and gradually expanded to include other numbers, while continuing to use 30, 60 and 90 as her favorites.
>
> --her patience in a one-step-at-a-time mode of operation was quite remarkable. Her format was quite stereotyped. (1) carry out one TURTLE step (turn, move or penup); (2) check to see if that looks right on the screen; (3) if so, write down the step and continue; (4) if not, clear the screen, retype all the steps previously written down and try another choice for the questionable one.
>
> --she rejected all suggestions that she incorporate new ideas into her work until she had thoroughly mastered the ideas with which she was already familiar.

In short, she created a very limited "microworld" of computer activities in which she could function successfully. She gradually expanded her microworld as she became confident of her mastery of it.

The microworld which Deborah chose for herself, the world of FORWARD 30 RIGHT 30, is very nearly as rich as all of Turtle Geometry. It includes squares, triangles, circles, "stars", "men", "rabbits", and a variety of abstract designs, as well as the mathematical concepts of perpendicularity, inverse operations, the Total Turtle Trip theorem, symmetry, similarity, estimation of lengths and angles, planning and debugging, and procedure writing. By repeating rotations of RIGHT 30, it is possible to turn the TURTLE 60, 90, 120, 180 or 360 degrees. Thus, Deborah could make a square, make the TURTLE reverse direction, and make simple symmetrical designs by using only 30 as an input to RIGHT and LEFT. Similarly, designs based on lengths of 30 TURTLE steps, fit nicely into larger units of 90 or 120 (30 + 90) TURTLE steps.

By limiting her inputs to numbers such as 30, 60 and 90, Deborah actually enhanced the possibility that her explorations would produce interesting results. We do not know why Deborah chose these numbers for her initial explorations, rather than numbers like 55, and 66, or 34, 45, and 56, that are often chosen initially by many other children. Deborah never appeared to calculate consciously that "three thirties make ninety," or that "three sixties makes the TURTLE reverse direction". Throughout her work she was unwilling to combine inputs, and never used an input greater than 100.

Deborah's work during the LOGO classes fell into three major phases: a period of severe dependence and insecurity; a period of exploration, creation of different designs and the gradual understanding of aspects of Turtle Geometry and LOGO programming; and a period of consolidating her learning by working on a major project which took her several class periods to complete. These three phases will now be examined in some detail.

2. Dependence and Insecurity

For the first seven classes, Deborah worked on "teaching the computer" to draw her initials. She had a strong need to have every detail of her work be "correct," and she seemed to be afraid of making mistakes. As a result, she needed reassurance from the teacher for everything she did, and was unsure of even the simplest details -- such as typing a carriage return at the end of every line. She was reluctant to write in her notebook, and easily lost track of what she had done successfully.

During this period she would wait patiently for the teacher's help, staring off silently into space for long periods of time, if he was busy elsewhere. She had good insights about where to move the TURTLE to produce the initials she was drawing but she had difficulty understanding how to translate those insights into LOGO commands. Although the project of drawing initials is an excellent introductory LOGO project for many children, it seemed too complex for Deborah. It clearly had a "correct" result, which Deborah needed to achieve. She would not allow herself to experiment or to try out something that might not prove to be "right".

During this period, however, she did develop some knowledge that she could use later on. While making the TURTLE draw the "M" for her last initial, she discovered that she could orient it correctly at each vertex by repeating turns of RIGHT or LEFT 30. She was exposed to the idea of a procedure, although she was not yet able to write one independently. She began to develop some facility with the keyboard. When her initials were finally complete, Deborah had a feeling of pride and "ownership," despite the large amount of assistance she had required to carry out the project. (See Figure 4.1)

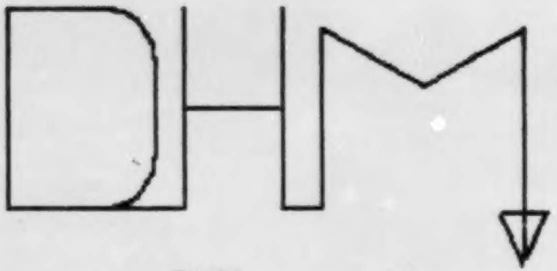


Figure 4.1

3. Exploration, and the Gradual Buildup of Skills and Confidence

Realizing that the initials project had brought out the most compulsive and insecure aspects of Deborah's personality, the teacher suggested a different approach, as soon as the project was finished. He "assigned" Deborah the task of "experimenting" with FORWARD, BACK, RIGHT and LEFT commands and the procedures RARC, LARC, RCIRCLE and LCIRCLE. Although Deborah was told to use a variety of inputs with these commands, during her first period of exploration, she only used two inputs: 30 and 90. She used 90 as an input for the RARC procedure, and used 30's for all other TURTLE commands.

Once Deborah got into "experimenting" mode, the compulsive need for success was eliminated, and she began to feel successful "without really trying". Deborah brought her copying skills into play, and developed a good way of writing procedures. Once she had a design she liked, she would look at the terminal, to find the last CLEARSCREEN command on the screen. Then she would copy into her notebook, all the steps following the CLEARSCREEN. If she had mistakenly used a LEFT 30, followed by a RIGHT 30 to correct it, she copied both, choosing to exercise no judgement as she copied. She was shown how to put a title at the top of the list of steps in her notebook, number each line, put the command END at the bottom, and type the entire procedure on the terminal.

Sometimes Deborah made mistakes in copying. When she did, she had two ways of correcting them: first she checked that the steps on the screen were copied correctly in her book; second, she checked that the steps in her book were copied correctly in a procedure. Her most common error was leaving out line numbers when typing her procedure. She could correct this herself, however, by retyping as much of the procedure as necessary. (She continued to increment line numbers by ones, until near the end of the classes.) When Deborah made a mistake, she would say "I goofed," in a wistful voice, and ask for help or reassurance. As she gradually came to realize that she had ways of fixing "goofs" by herself, she began to need help less and less.

Deborah also used the word "goof" in another context. When she was exploring freely, she would say "I'm just goofing around." There must have been an unstated connection in her mind between the two uses of "goof".

The approach used in teaching Deborah during this phase of her work was to show her no more than necessary to help her accomplish her purposes. She was shown EDIT (ED) and PRINTOUT (PO) to help with editing, and very little else for a long time. She needed to feel in control, and since she was carefully limiting the choices available to her, the teacher did the same. When an opportunity occured to show her something new, the teacher made limited suggestions, which she was encouraged to accept or reject. Deborah repeated successful activities over and over again until she was really secure with them, and ready to extend her microworld. She had control of whether to accept or reject suggestions, and gradually became able to accept them when she felt they were appropriate.

During this exploratory phase which lasted for seven or eight class sessions, Deborah created a number of interesting designs. SPYRO (see Figure 4.2) was her first independently initiated procedure. It grew out of exploratory work with the RARC subprocedure -- repeating RARC and increasing the input by 10.

```
TO SPYRO
1 RARC 20
2 RARC 20
3 RARC 30
4 RARC 30
5 RARC 40
6 RARC 40
7 RARC 50
8 RARC 50
9 RARC 60
10 RARC 60
11 RARC 70
12 RARC 70
13 RARC 80
14 RARC 80
15 RARC 90
16 RARC 90
17 RARC 100
18 RARC 100
19 RARC 100
20 RARC 10
END
```

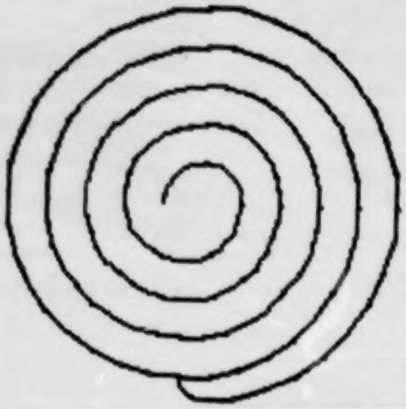
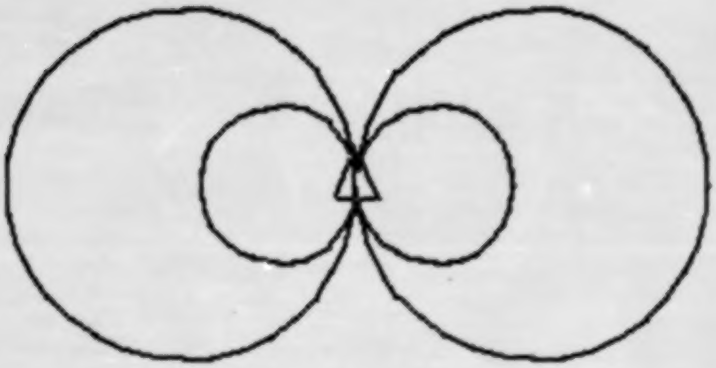


Figure 4.2

Another design became the procedure, EYES, made using RARC and LARC commands. (See figure 4.3).

```
TO EYES
1 RARC 90
2 RARC 90
3 RARC 90
4 RARC 90
5 LARC 90
6 LARC 90
7 LARC 90
8 LARC 90
9 RARC 40
10 RARC 40
11 RARC 40
12 RARC 40
14 LARC 40
15 LARC 40
16 LARC 40
17 LARC 40
END
```



EYES

Figure 4.3

Deborah's first use of a subprocedure as a "building block" came after she had made a CIRCLE procedure using four RARC 90s. Deborah made a FLOWER procedure by repeating a circle and a rotation. Deborah chose RIGHT 60 for the rotation, which led to a figure which closed after six repeats and made a symmetrical design (figure 4.4).

```
TO FLOWER
1 CIRCLE
2 RIGHT 60
3 CIRCLE
4 RIGHT 60
5 CIRCLE
6 RIGHT 60
7 CIRCLE
8 RIGHT 60
9 CIRCLE
10 RIGHT 60
11 CIRCLE
END

TO CIRCLE
1 RARC 90
2 RARC 90
3 RARC 90
4 RARC 90
END
```

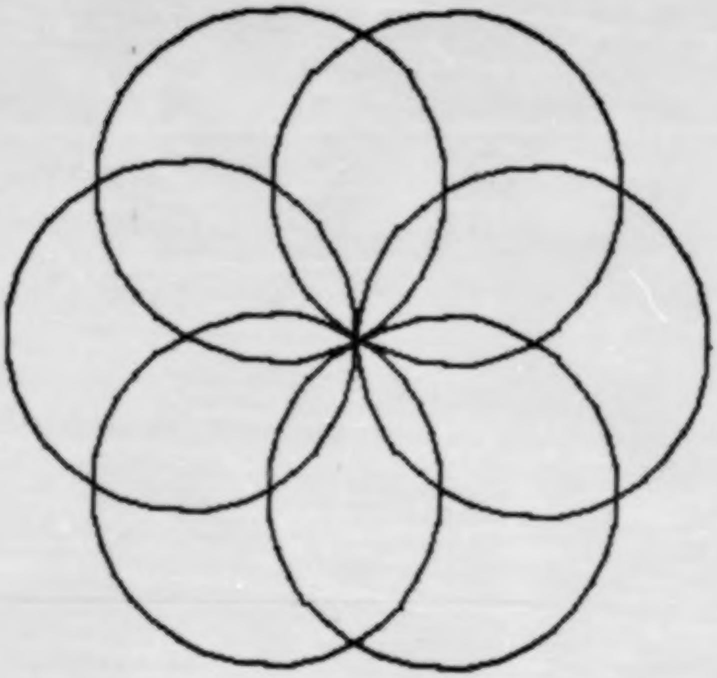


Figure 4.4

Deborah's most spectacular design was developed after the whole class watched a film which showed a number of computer designs. Among the designs shown was a six pointed star. When Deborah came right back to class from the film, she drew a six pointed star with the computer, without making a single mistake. She began by turning the TURTLE RIGHT 30, and proceeded to draw the star by using a combination of FORWARD 70s, and RIGHT 60s. Her choice of RIGHT 30 for the first step, and RIGHT 60 for the turns was critical in allowing her to carry out the project easily. Her strategy was to move the TURTLE forward 70 TURTLE steps, and then repeat RIGHT 60 until the TURTLE was aimed in the right direction. The total rotations needed are RIGHT 120 at the points of the star, and RIGHT 300 (equivalent to LEFT 60) at the inner vertices. Deborah did not seem to realize that she was <u>always</u> repeating RIGHT 60 two times and five times. At each point, she just kept turning the TURTLE until it was pointed in the right direction. At one point, Deborah missed the correct direction, and continued repeating RIGHT 60 for a total of 11 times until the correct orientation was achieved (Figure 4.5).

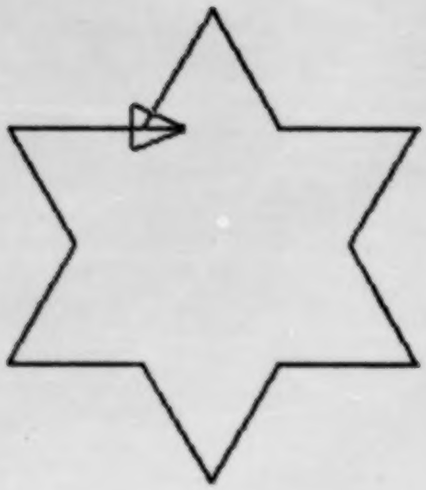


Figure 4.5

The teacher suggested an "easier" way for Deborah to teach the computer how to make the star. He suggested that Deborah teach the computer how to make <u>one point</u>, and then repeat that to make the star. Deborah accepted the suggestion, and taught the computer:

```
TO TRYANGLE
1 RIGHT 30
2 FORWARD 70
3 RIGHT 60
4 RIGHT 60
5 FORWARD 70
END
```

Since TRYANGLE included the step, RIGHT 30, it did not work when it was used as a subprocedure. Deborah noticed this when she tried to use it. The teacher helped her remove the extra step, and reminded her to use RIGHT 30 as the first step in her STAR procedure.

After Deborah had drawn the first TRYANGLE correctly, she was asked what command she had to give the TURTLE next. She looked at the situation carefully and after some time, said, "LEFT 60!" She then tried it and found that it worked. After that, she was able to build the star by repeating TRYANGLE, LEFT 60 until

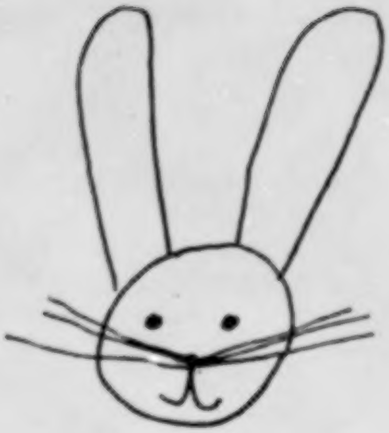
6 points were constructed. Here is her procedure, STAR, and the revised subprocedure, TRYANGLE:

```
TO STAR              TO TRYANGLE
1 RIGHT 30           2 FORWARD 70
2 TRYANGLE           3 RIGHT 60
3 LEFT 60            4 RIGHT 60
4 TRYANGLE           5 FORWARD 70
5 LEFT 60            END
6 TRYANGLE
7 LEFT 60
8 TRYANGLE
9 LEFT 60
10 TRYANGLE
11 LEFT 60
12 TRYANGLE
END
```

During this exploratory phase Deborah gained a great deal of the knowledge and confidence that seemed so difficult for her to obtain while trying to make the computer draw her initials. She learned to control the TURTLE, to write procedures and to use procedures as subprocedures in a "building-block" fashion. At the same time she learned to interpret error messages, correct her work, and to debug her procedures. Most important, she developed the confidence with the keyboard and the computer, that would enable her to carry out a major project.

4. Deborah's Major Project -- Drawing a Rabbit Head

In the sixteenth class session, Deborah indicated that she was ready to tackle a major project. She made a drawing of a rabbit (see figure 4.6a), and immediately said "It's too hard." A modification was suggested by the teacher using a square head, and triangular ears, that might be easier for her to carry out. (See figure 4.6b).



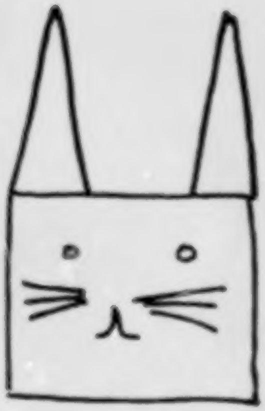
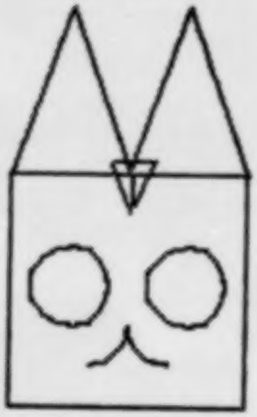


Figure 4.6a Figure 4.6b



RABBIT
Figure 4.7

Deborah began drawing the rabbit using direct commands. She built a square using TURTLE steps of 70 and 60 to make a side of length 130, and repeated FORWARD 70, FORWARD 60, for each side of the square. She then moved up the side of the square to make the eyes. When she got confused, she cleared the screen and tried again. When she hit a snag again, she cleared the screen again, and temporarily went on to another project -- rotating her FLOWER design and repeating it.

When Deborah came back to the rabbit project a few days later, she chose FORWARD 90, FORWARD 30, as the commands to make each side of her square. After several more attempts to make the computer draw the eyes and the nose, the teacher suggested breaking the problem into parts, and teaching each part to the computer separately. Deborah agreed to this and decided to teach the computer to draw the outside of the rabbit first. She could not think of a name for this, and finally decided to call it "HAT" (See figure 4.7a)

In the next session, Deborah added the eyes to the head, calling her procedure LITTLEEYES. When she began to teach LITTLEEYES to the computer, she intended to include, all the steps following CLEARSCREEN, as usual. The teacher explained that HAT and LITTLEEYES were both part of RABBIT, but that HAT should not be part of LITTLEEYES.

Her strategy of using 30 as an input helped her locate the eyes sucessfully in the head. The sides of the head were now 120 units long, and since she moved the TURTLE across the head in steps of 30 units, she was able to center the eyes with no difficulty. (See figure, 4.7b)

When Deborah began to work on the nose, she got really stuck. She had two different problems to resolve: How to make the nose, and where to put it. The teacher showed her how to separate the two problems, and make the nose all by itself -- forgetting about the rest of the rabbit for the time being. Then, once a good nose had been made, she could work on placing the nose in the right position.

Deborah planned to make the nose as shown in figure 4.6. The teacher showed her how to start with RARC, turn the TURTLE all the way around, come back to the beginning using LARC, and then reverse the whole process to make the other side. Deborah understood the idea, but needed help working it out. Whenever Deborah needed to turn the TURTLE around she used RIGHT 90, RIGHT 90. (See figure, 4.7c)

Later, when she began to make the rabbit's ear, she again demonstrated her accuracy in choice of inputs. She had moved the TURTLE to the top of the rabbit's head. Deborah needed three tries to make the first ear. The key decision was how far to turn at the top. Since she had turned the TURTLE RIGHT

20 at the base of the ear, a turn of RIGHT 140 was needed to make an isoceles triangle. In Deborah's first try, she turned RIGHT 90 four times, then RIGHT 90, RIGHT 30, RIGHT 30, for a total of 150 degrees. The far end of the ear did not line up easily with the top of the head. Her second try was RIGHT 900 (mistake), followed by two RIGHT 90s, to straighten the TURTLE out again, followed by RIGHT 70, RIGHT 20, RIGHT 20, RIGHT 20, RIGHT 20, for a net rotation of 150 degrees again. Once again, she had trouble lining up the far end. On the third try, she turned RIGHT 90, RIGHT 30, RIGHT 20, for a total rotation of 140 degrees. This made it easy to line up the far end of the ear, which happened to come out almost exactly at the far end of the head. At this point, Deborah wrote out all the steps, and said "I can do the same thing on the other side. Should I give this a name?" (See figure 4.7d). She then defined the procedure TO EARS.

At the next session, Deborah was helped to separate the steps that made the ear, from the steps that set it in position. By doing so, she could use the same subprocedure, EARS, to make ears on both sides. Her completed RABBIT is shown in figure 4.7.

At one point, while working on the ears, Deborah rejected help from the teacher, asserting her independence by proclaiming loudly, "I know what I'm doing!"

During the RABBIT project Deborah mastered some aspects of LOGO and used other concepts that she had not yet mastered. Some of Deborah's learnings were:

> --first and foremost -- the idea of using subprocedures to break a large project into a group of small projects.
>
> --that a subprocedure can have two parts, drawing the object (nose, ear) and locating it, and that the same subprocedure (EARS) can be used in more than one place.
>
> --that with patience, even complicated problems can be worked out by step by step using a trial and error approach (location of nose, location and shape of ears).
>
> --that 90, 30 and 20 are really good numbers to use in combinations (In her entire RABBIT procedure Deborah used a total of 75 procedural steps. Forty-nine out of fifty-eight TURTLE commands used inputs of 90, 30 or 20.)
>
> --for the first time, she developed complete confidence in her ability to understand what she was doing ("I know what I'm doing!") while

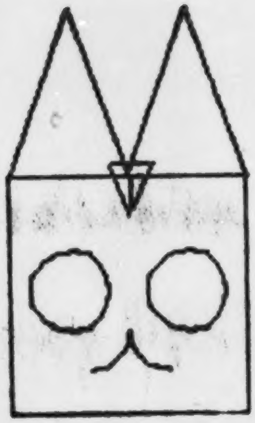


Figure 4.7

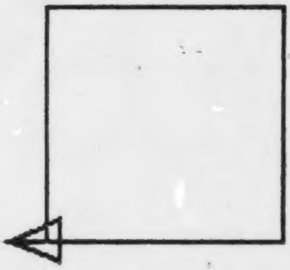


Figure 4.7a

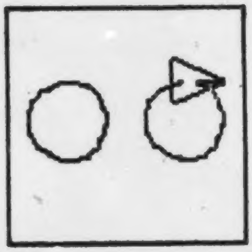

HAT LITTLEEYES

Figure 4.7b

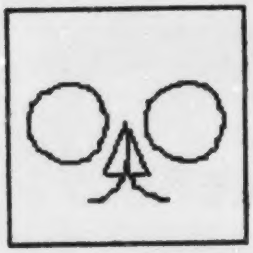

~~HAT~~ LITTLEEYES FACE

Figure 4.7c

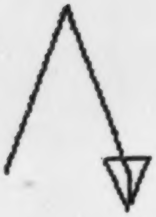


Figure 4.7d

> recovering from the frequent errors she made. Perhaps this was really the most important learning for Deborah -- not just to be in control of a learning environment, but to know she was in control, to feel a sense of mastery. To be able to learn from and not be daunted by her own mistakes.

With the completion of her rabbit project, Deborah had almost totally reversed her initial feelings of dependence and incompetence. She invited her parents, teachers and school principal to visit the computer lab, and in many ways, demonstrated to her visitors and classmates her new found sense of confidence, satisfaction and power.

Here is an annotated copy of the procedures that Deborah developed to solve the problem of drawing a "rabbit" with the computer.

```
TO RABBIT            RABBIT is Deborah's superprocedure
5 HAT                HAT draws the outside of the head
10 LITTLEEYES        LITTLEEYES draws the eyes
15 FACE              FACE moves from the eyes to the
20 PENUP             center of the head and draws the nose
25 FORWARD 70
30 FORWARD 3         move to the top of the head to draw an ear
35 RIGHT 20          orients the TURTLE to draw an ear
40 PENDOWN
45 EARS              draws one ear
50 RIGHT 90
55 FORWARD 50                                 (See Figure 4.7)
60 FORWARD 3
65 FORWARD 60        moves over to draw the second ear
70 FORWARD 5
75 FOWARD 3
80 RIGHT 90
85 RIGHT 20          orients the turtle to draw the second ear
90 EARS              draws the second ear
END
```

```
TO HAT              (draws a box to make the outside of the
1 FORWARD 90        rabbit's head)
2 FORWARD 30
3 RIGHT 90                                   (See Figure 4.7a)
4 FORWARD 90        HAT draws a box of size 120 by
5 FORWARD 30        repeating FORWARD 90 FORWARD 30
6 RIGHT 90          to make each side of the box
7 FORWARD 90
8 FORWARD 30
9 RIGHT 90
10 FORWARD 90
11 FORWARD 30
END

TO LITTLEEYES       (draws eyes of rabbit)
1 RIGHT 90
2 FORWARD 70   ┐
3 FORWARD 12   ┘    moves up the side of the head
4 RIGHT 90                                   (See Figure 4.7b)
5 PENUP
6 FORWARD 30        locates the first eye
7 PENDOWN
8 RARC 20      ┐
9 RARC 20      │    draws the first eye
10 RARC 20     │
11 RARC 20     ┘
12 PENUP
13 FORWARD 30  ┐    moves across the head to
14 FORWARD 30  ┘    locate the second eye
15 PENDOWN
16 RARC 20     ┐
17 RARC 20     │    draws the second eye
18 RARC 20     │
19 RARC 20     ┘
END
```

```
TO FACE              (locates and draws nose)
1 RARC 20
2 RARC 20            moves to bottom of eye
3 PENUP                                     (See Figure 4.7c)
4 FORWARD 20
5 FORWARD 10         moves to center of head
6 RIGHT 90
7 RIGHT 90           orients TURTLE straight down
8 RIGHT 90
9 PENDOWN
10 NOSE              draws nose
END

TO NOSE              (draws nose of rabbit)
1 RARC 20            draws an arc
2 RIGHT 90 ┐
3 RIGHT 90 ┘         reverses the TURTLE
4 LARC 20            traces back over previous arc
5 RIGHT 90 ┐
6 RIGHT 90 ┘         reverses the TURTLE
7 LARC 20            draws second arc
8 RIGHT 90 ┐
9 RIGHT 90 ┘         reverses the TURTLE
10 RARC 20           retraces arc, to return
END                  the TURTLE to its starting position

TO EARS              (draws one ear)
1 FORWARD 90         draws the left hand side of ear
2 RIGHT 90
3 RIGHT 30           turns TURTLE total of 140 degrees
4 RIGHT 20                                  (See Figure 4.7c)
5 FORWARD 70
6 FORWARD 20         draws second side of ear
7 RIGHT 20           orients TURTLE straight down
END
```

5 . Dennis

5.1 Dennis' Working Style

Dennis is a bright, articulate student who makes connections easily, and has a strong desire to continually expand his knowledge. He began by being very excited about using the computer, asking a lot of questions about what the computer could do, and how to accomplish different things. Dennis had a lot of difficulty setting into a particular project. Although he quickly progressed through a number of LOGO activities, he seemed to have difficulty with syntax -- remembering how to WRITE his file, or use a PRINT statement, or incorporate variables into his work.

Dennis seemed to understand a concept rather quickly, but to have difficulty with the details involved in using the concept. He liked to keep all his work "in his head," and only gradually accepted the idea of taking notes that might help him with his work. Notes, once taken, were often left lying around the room. Having worked out a series of steps to draw a particular figure on the screen, he seemed to have difficulty concentrating on the task of accurately translating those steps into a LOGO procedure.

Dennis eagerly sought new information and asked questions about new ideas. On the other hand, he did not like to ask for help when he encountered difficulties in his work -- and often rejected "helpful suggestions," when they were offered. He seemed to prefer his own independent, ideosyncratic ways of doing things, to learning more efficient ways from another person. When he was asked, in a pre-LOGO interview, "If you're stuck on something in school, what kinds of things do you do?" Dennis answered: "I try to control myself to work harder on it, and if worse comes to worse (sic) I just sit in the corner and sulk. I just try to deal with it." This approach to situations in which he was "stuck" was quite characteristic of his work in the LOGO classes.

One of the areas in which Dennis often had difficulty was taking a series of successful steps already worked out on the screen, and translating them into a LOGO procedure. He persisted for a long time numbering steps by ones, *despite the fact that he understood quite well the justification for numbering steps by fives or tens*. Since he often made mistakes copying steps, he often had to make laborious changes in his procedures. To compensate for the problem of forgetting line numbers, Dennis usually put many steps on *one* line. He persisted in using this technique despite numerous errors which required that he re-type long lines of commands. It was more important for Dennis to use *his* method (numbering by ones, and putting many steps on a line) than it was for him to have a method that made debugging easier (numbering by tens, putting one command on each line, and carefully keeping track of the line number).

It seemed as though Dennis expected that bugs would not occur -- even though he continued to make small errors throughout his LOGO classes -- and that he did not have to work in a way that made debugging easier. The irony was that he understood the advantages, of the suggested methods, but persisted in using his own, even though they were less successful, and caused him more problems later. "It's my funeral," he once remarked in such a situation.

It is sriking to see a bright child persist in strategies known to be unsuccessful, when alternative strategies are available.

Dennis seemed to be at loose ends during the first few weeks of computer classes. He explored a lot of different ideas, without settling in to work in one particular area, or committing himself to a project. Finally, after a great deal of coaxing, he agreed to work on a project which the teacher suggested -- a moving car with spinning wheels. He worked on this project for eight classes. Although he had a lot of difficulty carrying it out (see below), and was frustrated at several points, he maintained a strong commitment to the project until it was completed.

5.2 Dennis' Work in Turtle Geometry With Triangles and Variables

Dennis' work in Turtle Geometry was characterized by an excellent understanding of the use of LOGO to create designs, by a good ability to estimate distance, and a lot of problems with orientation and estimating angles.

Dennis worked on a series of short projects involving triangles. He started with drawing a triangle, and continued to make an equilateral triangle, variable sized triangles, designs using triangles, and a design consisting of nested triangles. This work occupied five different sessions mostly during the early classes, but including one class near the end.

He constructed his first triangle using a trial and error approach. For his first attempt at drawing an equilateral triangle he used rotations of 128 and 110 degrees at the corners. When he copied his steps to make a procedure, TRI, he made an arithmetical error in combining angles, and wound up with the shape shown in Figure 5.1.

```
TO TRI
1 RIGHT 140
2 FORWARD 99
3 LEFT 88 - - - - (erroneous step)
4 BACK 99
5 LEFT 140
6 RIGHT 70
7 FORWARD 99
END
```

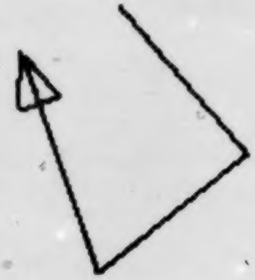


Figure 5.1

Although Dennis realized that this was not what he expected, he drew on previous experience in making a recursive design to create the patterned procedure ULTRATRI:

```
TO ULTRATRI
1 TRI
2 RIGHT 80
3 ULTRATRI
END
```

Returning to constructing a triangle, Dennis abandoned equal lengths, and produced a close approximation to a triangle. He called his procedure THRI (to distinguish it from TRI).

```
TO THRI
1 RIGHT 140
2 FORWARD 99
3 RIGHT 130
4 FORWARD 110
5 RIGHT 120
6 FORWARD 93
END
```

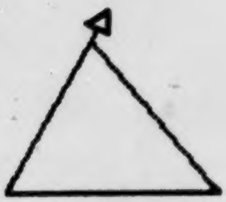


Figure 5.2

At a subsequent class, Dennis asked how to make a triangle of variable size, so that he could make a design of nested triangles. The teacher explained that it would be simpler to use variables neatly if he could make an equilateral triangle, and that an equilateral triangle had equal angles at each point. With this information, Dennis was able to find the correct angle, 120 degrees, in five or six tries. Dennis called his new procedure THRI1, and his variable, "S3.

```
TO THRI1:S3
1 FORWARD :S3
2 RIGHT 120
3 FORWARD :S3
4 RIGHT 120
5 FORWARD :S3
6 RIGHT 120
END
```

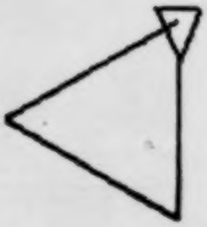


Figure 5.3

Next, Dennis began to make a set of nested triangles using the variable-sized triangle he had already created. He wanted to make a design that looked like this:

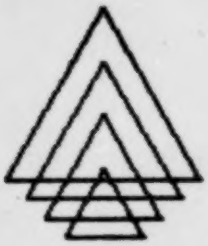


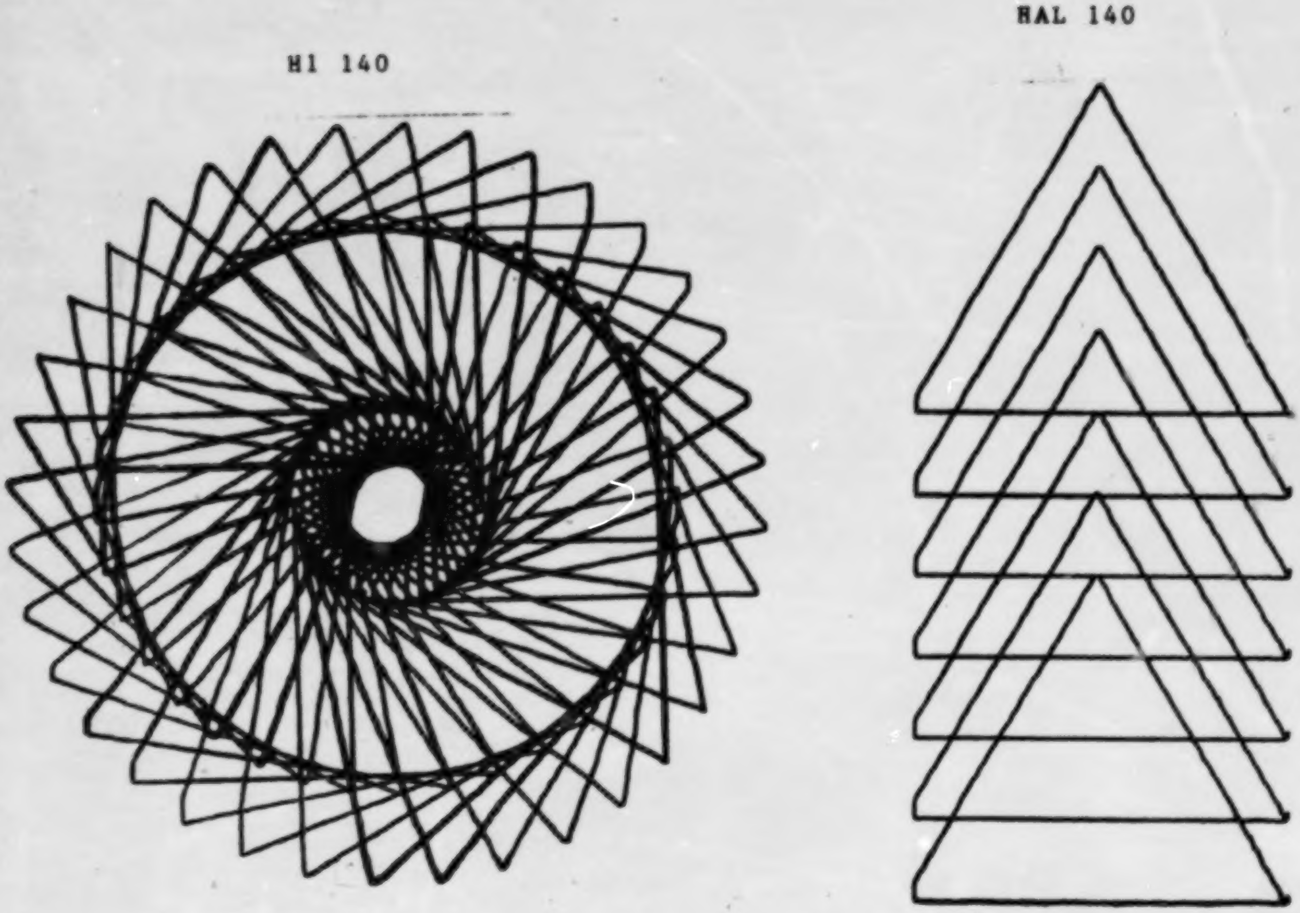
Figure 5.4

While making it, he had a lot of trouble with the state of the TURTLE: orientation of the TURTLE before and after making the triangle; re-orientation of the TURTLE to move it vertically before making the next triangle; whether the pen was up or down, etc. After struggling through several repetitions, he wanted to give up.

In the next class, he was reminded that he could make a procedure to repeat the group of steps that kept recurring, and he defined these procedures:

```
TO SECT:SIZE          and          TO HAL :SIZE
1 LEFT 90                          10 SECT :SIZE
2 THRI1:SIZE                       20 HAL :SIZE
3 RIGHT 90                         END
4 PENUP
5 BACK 30             and          TO H1 :SIZE
6 PENDOWN                          10 SECT :SIZE
END                                20 RIGHT 50
                                   30 H1 :SIZE
                                   END
```

He was content to leave these procedures as they were without varying the size, but enjoyed the effects of a variety of inputs.



HI 140

HAL 140

Figure 5.5

At a later point, Dennis came back to his variable triangle procedure, and followed through with his original intention of making a set of nested triangles. At that point he had a clearer idea of how to make use of variables, and a good strategy for solving the nested triangle problem. He realized that if he could solve the problem of nesting a second triangle at the proper position inside the first triangle, that the complete nested design could be drawn by repeating the same process over and over.

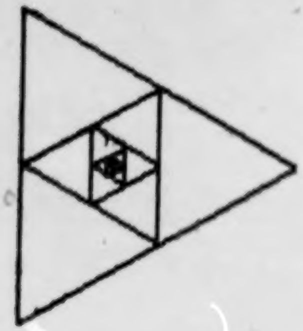


Figure 5.6

Dennis realized that the essence of the process was to draw one triangle, move the TURTLE forward half the length of the triangle, and rotate the TURTLE, before repeating the same process with a second triangle, half the size of the first.

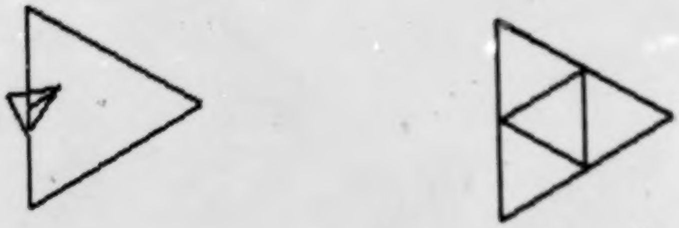


Figure 5.7

He had a problem determining how far to rotate the TURTLE before drawing the second triangle. Resolution of this problem required a great deal of trial and error with careful attention to detail. After about ten tries he was ready to settle on a rotation of 55 degrees, as the correct amount to rotate the TURTLE. When he looked carefully at the resulting figure, he was able to see that the two triangles were slightly out of alignment. His next attempt, 60 degrees, was the correct rotation to produce the effect he was seeking.

At this point he was able to talk through the necessary steps.

"THRI1 :SIZE
move forward half of :SIZE
turn RIGHT 60 degrees,
repeat the whole process with a value of one half of :SIZE."

His procedure became:

```
TO Q :SIZE
10 THRI1 :SIZE
20 FORWARD :SIZE/2
30 RIGHT 60
40 Q :SIZE/2
END
```

Dennis wanted to make Q stop, "when :SIZE is less than 10." After some additional conversation about the location of the stop rule in the procedure, he added line 35 to Q:

```
35 IF :SIZE < 10 STOP
```

which had the desired effect.

Some things to notice about Dennis' work with triangles and variables: he consistently had difficulty with problems of TURTLE orientation, as in making his HAL figure or in nesting the smaller triangle inside the larger one. He had no difficulty with the <u>idea</u> of variables -- which he related to something he was familiar with from math class. But he had difficulty with the syntax of using variables -- where to put the dots, in both the commands and in the procedure titles. He seemed to understand the way in which his procedures functioned, and was able to see how his descriptions of what he wanted to do translated into LOGO. He never did carry through a project with variables entirely on his own.

5.3 Dennis' CAR Project

In class thirteen, the teacher insisted that Dennis commit himself to a long term project. He chose to draw a moving car, with spinning wheels. Once committed to the project, he followed through on it for eight classes -- despite a good deal of frustration with the details.

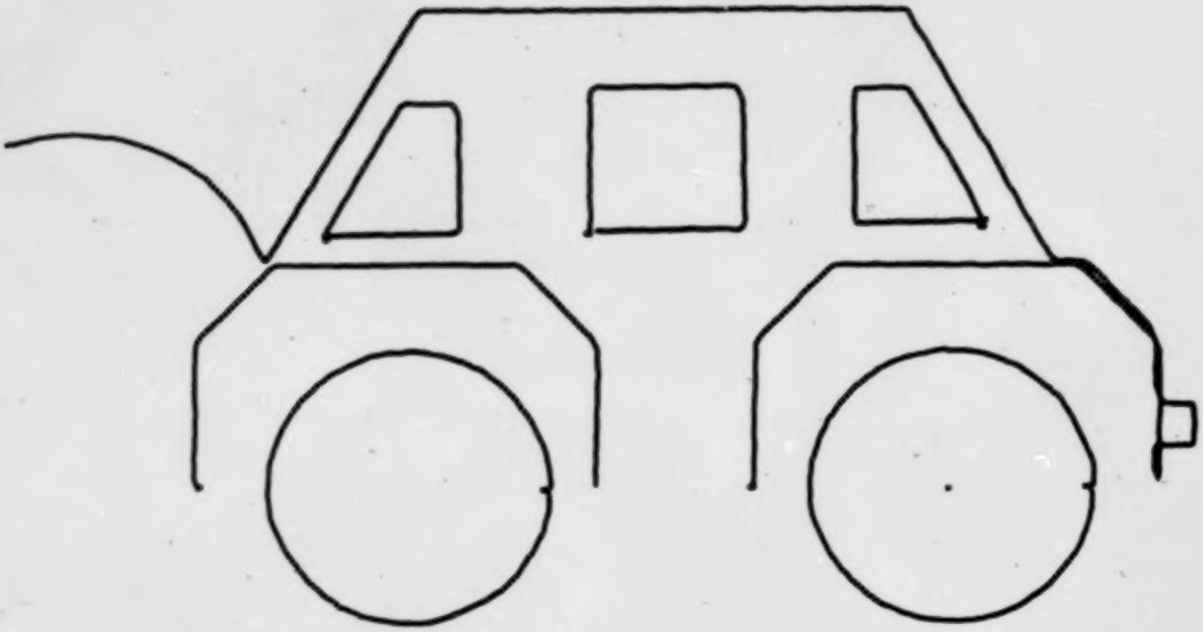


Figure 5.8

Rather than describe Dennis' work on this project sequentially, we differentiate between those aspects of the work that were easy for him, and those that were difficult. We offer some speculation about why so much of this project, which would have been simple and straightforward for a student like Jimmy was so difficult for Dennis.

Most of Dennis' problems seemed to be in the area of "work habits" and attention to detail. Lack of attention to detail, combined with left-right orientation problems, meant that a lot of Dennis' difficulties showed up as TURTLE state bugs.

On the other hand, Dennis rather easily understood the important "concepts" of the project: how to use positive and negative inputs to the SPIN command, together with a centered circle procedure, to make a wheel that spins around its own center; how to use MOVET to move the car; how to use 45 and 60 degree angles to make the fenders and body of his car; and how to use symmetry to make his front window a "reversed" version of his rear one. During the course of the project he learned to make use of separate subprocedures for each part of his design, and began to number the steps in his procedures by fives.

In illustration of both the strengths and weaknesses of Dennis' work on this project, consider the process by which Dennis made the windows for his car, which took three class periods to complete.

At the end of class #15, Dennis had completed a series of procedures that drew the basic outline of the car as shown in figure 5.9. He wanted the rear window located parallel to the body of the car.

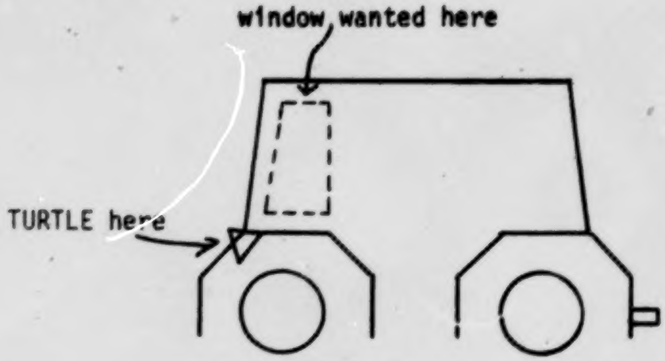


Figure 5.9

Two suggestions were made to help him get started. First, that he draw the window by itself on an empty screen first, and add it to the car as a separate procedure later; second, that he look at the BODY procedure, to find the angle to use for the windows.

Looking at the BODY procedure, Dennis quickly noticed that the correct angle was 60 degrees. He then cleared the screen and easily drew a window, starting with the TURTLE in the upright position:

```
TO WIN1
1 RIGHT 30 FORWARD 60 RIGHT 60 FORWARD 20 RIGHT 90 FORWARD 50
2 RIGHT 90 FORWARD 50
END
```

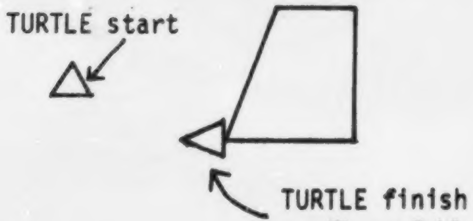


Figure 5.10

With the TURTLE in the vertical position, Dennis had no difficulty turning it right 30 degrees, going forward and then turning it 60 degrees to make it horizontal. It seemed as though he made use of the fact that 30 and 60 add up to a full 90 degrees in deciding to use 60 for the second turn, although this was never stated. The distance estimates were also fairly easy for him to make. The WIN1 procedure left the TURTLE facing horizontally to the left.

When Dennis attempted to place the window in the car, however, he did not realize that he had to orient the TURTLE vertically before using his WIN1 procedure. As a result, the window kept coming out in the wrong position and Dennis became very confused.

When he came back to working on the window, he was reminded that to orient his window properly he had to begin with the TURTLE in an upright position. He had decided to add another feature to the car, before drawing the window. A curved antenna was added, completing the body, so that the turtle, now in position A, needed to move to position B (shown in figure 5.11).

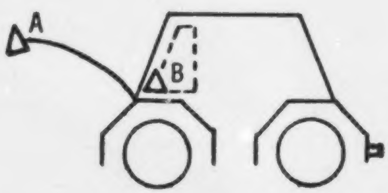


Figure 5.11

After experimentation, estimation of distances, etc., Dennis wrote the following procedure:

```
TO WIP (window in place)
1 MO HUBS BUM BODY ANT PENUP RIGHT 180 FORWARD 100 RIGHT 90 FORWARD
      80 RIGHT 180 RIGHT 20 FORWARD 10 PENDOWN
2 WIN1
END
```

This procedure accurately placed the turtle in an upright position before drawing the window. The subprocedure MO moved the car over, so that the entire picture fit on the screen. The other subprocedures drew parts of the car. In typing the steps Dennis originally left out two commands: ANT and PENDOWN. He had to retype the entire procedure twice to replace them. He placed all the steps on one line despite several discussions explaining how numbering steps by 10s and placing each step on a different line would make further corrections and debugging easier.

At the end of the period Dennis forgot to save the procedure on his disk by writing a file! He had to repeat the whole process again next time -- including the exploration, since he had kept no notes. This time, he came out with a slightly different location for the TURTLE -- one which left the TURTLE tilted 1 degree from the vertical.

```
TO WIP
10 MO HUBS BUM BODY ANT
20 RIGHT 180 PENUP FORWARD 100 RIGHT 90 FORWARD 60 FORWARD 10 RIGHT 70
      RIGHT 180 LEFT 45 BACK 5 LEFT 6 PENDOWN
30 PENDOWN WIN1
END
```

Dennis' final version of WIP shows some improvement over what he had previously done. The lines are numbered by tens. Each line of WIP has a distinct purpose. Line 10 draws all the previous parts of the car; line 20 moves the TURTLE over to draw the window; line 30 draws the window.

He had no problem figuring out how to draw the middle window as a square:

```
TO WIN2
10 PENUP BACK 100 RIGHT 90 PENDOWN FORWARD 55 RIGHT 90
    FORWARD 55 RIGHT 90 FORWARD 55 RIGHT 90 FORWARD 55
END
```

He originally made WIN2 draw a square with sides of 43. When he decided to change to a size of 55, he had to retype all of line 10.

Before Dennis began to draw WIN3, The teacher suggested that he make use of symmetry by using the WIN1 procedure as a model, and draw the new window by reversing the turns in WIN1, while keeping all the distances the same. Dennis understood this approach and decided to use it. Once again he forgot that the TURTLE had to be vertical before carrying out this aspect of the project. Finally he recovered after being reminded to set the TURTLE heading up before reversing the steps of WIN1. Finally Dennis produced the procedure WIN3:

```
TO WIN3
10 PENUP BACK 145 PENDOWN RIGHT 90 LEFT 30 FORWARD 60 LEFT 60
      FORWARD 20 LEFT 90 FORWARD 50 LEFT 90 FORWARD 50
END
```

Once again, Dennis left out PENDOWN in copying the steps, and had to redo the whole line. The underlined steps are the steps of WIN1 reversed, although they are not set off as a separate sub-procedure, or even accorded a line of their own in WIN3.

Throughout this process Dennis was constantly frustrated by his failure to remember where the TURTLE had to be heading at the start of a procedure like WIN1; by his failure to take useful notes -- leading to the omission of steps like PENDOWN; and by his method of including as many steps as possible on one line -- necessitating the retyping of an entire line, everytime he made a typing error or omission. Although all these problems were pointed out to him by the teacher, along with suggestions for how to alleviate them, Dennis seemed to prefer to struggle along with his own methods rather than to 'bother' to take notes or put each step on its own line in a procedure.

5.4 A Parallel Between Two Projects, and a Question

An interesting parallel may be observed between details that occured in each of two projects described: making the window for his car, and drawing the nested triangles.

In drawing the window Dennis started with the TURTLE in an upright position. He turned it RIGHT 30, then moved FORWARD 60 and then turned the TURTLE RIGHT 60, to make the top of the window. All this was done without any explanation -- following a discussion in which Dennis decided to make the window slant at an angle of 60 degrees, to parallel the BODY of his CAR (see figure 5.12a).

A similar situation occurred when he was making the nested equilateral triangles. The first triangle started in an upright position. After completing one triangle, Dennis moved the turtle forward half of its length, and once again, had to turn the turtle right 60 degrees. In this case, he did not "know" that the angle to turn was 60 degrees, and had to establish it by means of a trial and error process of successive approximations. (see figure 5.12b).

The question is -- why was Dennis not able to make use of the knowledge he had about 60 degree angles in drawing the nested triangles, when he was able to use it in drawing the car window? Possibly he used the fact that 30 + 60 = 90, when drawing the window. It is interesting to conjecture that he might have been able to find the correct angle without experimentation if the first triangle had been oriented at an angle of 30 degrees to the vertical to start with (see figure 5.12c).

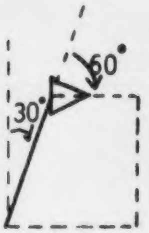


Figure 5.12a

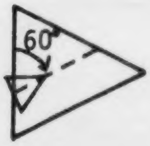


Figure 5.12b

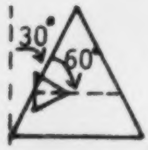


Figure 5.12c

5.5. Dennis' work With Dynaturtle

Dennis' work with Dynaturtle gave an opportunity for him to discover new strategies to solve problems in another domain. The Dynaturtle is a simulated "Newtonian turtle" which follows Newtonian laws of motion, rather than regular TURTLE commands (see Part II, Chapter 6). The motion of the dynaturtle can be changed only by giving it a "kick" (impulse) in a given direction. Dennis was given two games to use with the dynaturtle: GAME1 required him to land the dynaturtle on a fixed target; LUNER required him to move a "falling" dynaturtle so that it landed on a specified "landing pad", rather than "crash" on the moon. (See Part II, Chapter 6 of this report for more information about dynaturtle activities.)

Dennis worked on dynaturtle activities for all or part of three different classes. He gradually began to show a lot more interest in the messages printed by the different game programs than in playing the games themselves. (See section 5.6, below.) Dennis' work with the LUNER game was characteristic of his approach to dynamics. The LUNER game begins with the turtle situated above, and to the left of a platform on which it must land for a successful outcome. The initial orientation of the turtle is straight up -- as shown in fig. 5.13. If the dynaturtle is not kicked, it will "fall" straight down and "crash". (The LUNER program gives the dynaturtle a steady downward "gravitational "kick."

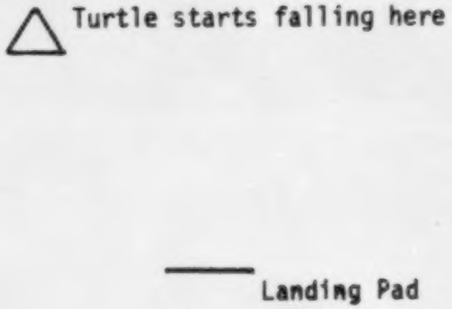


Figure 5.13

One fairly simple strategy for the game would be to quickly turn the dynaturtle until it was facing horizontally to the right, then give it three or four kicks. The combination of rightward velocity and continuing downward acceleration would be likely to land it on the pad -- depending on the timing and number of the rightward kicks.

Dennis quickly realized that if left alone, the dynaturtle would fall to the bottom of the screen and crash. He discovered that he could counteract this by kicking the dynaturtle straight up, temporarily counteracting the gravitational effect, and giving the dynaturtle an upward velocity. Most of the time he gave the dynaturtle so many upward kicks that it would rapidly rise off the screen (wrapping around and coming onto the screen again from the bottom.) By the time he started to turn the dynaturtle to the right, it was so far above the landing pad that a slight rightward motion would cause it to crash far to the right of the pad.

During his first two days of working with LUNER, Dennis and Harriet worked together. She had adopted a strategy of turning the dynaturtle so that it was aimed at the landing pad, then kicking several times. This usually had the desired effect of landing in the right place, although at unneccessarily high speed. Although Dennis watched Harriet at work he did not adopt her strategy when she was present. When he worked alone, however, he did adopt a version of Harriet's method and was able to be reasonably successful with the LUNER game.

Throughout his work with dynaturtle, it was not unusual for Dennis to type the K, R, and L keys very rapidly (kick, right and left), without looking at the effects of any of his actions before typing new instructions.

5.6 Dennis' Work with Printing, Messages and the Beginnings of Interactive Programming

Dennis showed a lot of interest in using the computer in an "interactive" fashion. Although he did not undertake a major interactive project, his continued interest indicated this as a possible future direction for him. From time to time, in the early stages, he asked about uses of the computer other than Turtle Geometry, and was shown a little bit about how to use PRINT, arithmetic commands, and simple list processing. He limited his initial work with PRINT to making a design of a fa[illegible]

Dennis devoted one entire class to using the computer as a typewriter/editor for a story he was writing about Julius Caesar. He did this by writing a procedure with 53 lines, each of which printed a sentence. In the course of this task, which took about two hours, Dennis also mastered the use of Control keys for editing.

While he was working with dynaturtle, Dennis became interested in the messages printed by the computer at the end of the game. These messages told the player whether the landing had been soft, rough or a crash, depending on the dynaturtle's speed at the moment of impact. He complained that these messages were "dumb", and the teacher suggested that he change them. He stepped through the LUNER procedure, and all its subprocedures, until he found the subprocedure FINISH, which printed the messages. Dennis spent almost all of one period creating his own rather elaborate version of these messages:

```
TO FINISH
10 IF SPEED < 7 PRINT [WOWEEEE!!!!YOU FINALLY DID IT MY GOD YOUR SPEED
WAS GOOD] PRSPEED STOP
20 IF SPEED < 25 PRINT [YOU ROTTEN LOUSY NO GOOD PLAYER YOUR SPEED WAS
SO FAST I'M SURPRISED I'M NOT PEELING YOUR BODY OFF OF THE LANDING PAD A
   SECOND GRADER WITH THE BRAINS OF AN AMOEBA COULD DO BETTER THAN YOU JUS
T DID !!!!! TRY AGIAN DUMMY!!!] PRESPEED STOP
30 PRINT [HA!HA!HA! YOU STUPID IGNORANT DUMP REALLY DUMB PLAYER!!!!
YOU DIDN'T LISTIN TO ANYTHING I SAID YOU CRASHED!!! YES YOU CRASHED!!!
COM'ON NOW KLUTZO TRY AGIAN HA!HA!HA!HAAAAA!!! HEH HEH HEE HO!!!!
!!HEH HEH HEH HEH !!!!] PRSPEED STOP
```

Throughout his work on thse messages, Dennis appeared to be more deeply involved with his work than he did on Turtle Geometry projects. He laughed,

```
TO GJC
1 PRINT ["THE COOL HEADED MAN OF GENIUS WITH AN ERRATIC VEIN OF]
2 PRINT [SEXUAL EXUBERANCE, UNDOUBTEDLY CHANGED THE COURSE]
3 PRINT [OF HISTORY AT THE WESTERN END OF THE OLD WORLD", ]
4 PRINT [A QUOTE FROM THE ENCYCLOPEDIA BRITANNICA BEST]
5 PRINT [POTRAYS GAIUS JULIUS CAESAR ON THE WHOLE. ]
6 PRINT [ ]
7 PRINT [CAESAR DID A LOT OF THINGS IN HIS TIME, EVERYTHING]
8 PRINT [FROM WRITING HIS "COMMENTERIES" TO CAMPAIGNING IN]
9 PRINT [GAUL, BRITIAN, AND MANY OTHER PROVINCES. HE UPDATED]
10 PRINT [THE ROMAN CELENDAR WITH HIS JULIAN CALENDAR THAT IS QUITE]
11 PRINT [ACCURATE. CAESAR IS SAID TO HAVE BEEN BORN IN 100 B. C. ]
12 PRINT [THE 12 OR 13 OF JULY; HE WAS BORN IN THE ROMAN MONTH]
13 PRINT [QUIATILIS THAT WAS CHANGED TO JULY IN HIS HONOR. ]
14 PRINT [ ]
15 PRINT [CAESAR WENT INTO POLITICS AS THE COURSE OF THINGS]
16 PRINT [, AND HIS EXPECTATOINS OF GOING TOO FAR AT FIRST]
17 PRINT [WERE NOT TOO HIGH. IN 69 B. C. CAESAR WAS VOTED]
18 PRINT [QUEASTOR (FIRST STEP IN ROMAN POLITICS). IN]
19 PRINT [AROUND 63 HE WAS VOTED AS A HIGH PRIEST, AND]
20 PRINT [IN 62 CAESAR WAS MADE PRAETOR (HE WISHED HE]
21 PRINT [GO HIGHER). AND THEN IN 60 HE TRIED FOR THE]
22 PRINT [COUNSULSHIP AND WON. ]
23 PRINT [ ]
24 PRINT [IN 84 BEFORE POLITICAL POWER CAESAR MARRIED CORNELIA]
25 PRINT [, A DAUGHTER OF LUCIUS CORNELIA CINNA (A NOBLE OF]
26 PRINT [GAIUS MARIUS'S ASSOCIATES). BUT LCIUS CORNELIUS SULLA]
27 PRINT [(A POLITICAL FIGURE IN ROME AND A VICIOUS OPPONENT]
28 PRINT [OF MARIS'S) ORDERED THE DIVORCE OF CAESAR AND CORNELIA]
29 PRINT [CAESAR DIDN'T LISTEN TO SULLA AND FOUGHT IN ASIA FOR]
30 PRINT [THE ROMAN ARMY. HE THEN CAME BACK WHEN IN 78 SULLA DIED]
31 PRINT [, THAT WAS THE SAME YEARHIS WIFE DIED. ]
32 PRINT [ ]
33 PRINT [".... AN ERRATIC VEIN OF SEXUAL EXUBARENCE", MEANING]
34 PRINT [HE PROBABLY DIDN'T ONLY FOOL AROUND WITH HIS WIFE. ]
35 PRINT [IT APPEARS THAT CAESAR'S SEXUAL ESCAPDES WERE QUITE]
36 PRINT [FAR FETCHED EVEN FOR GREEK AND ROMAN STANDARDS. ]
37 PRINT [HE DANGEROUSLY HAD AFFIARS WITH POMPEY'S WIFE , AND]
38 PRINT [CLEOPATRA THE QUEEN OF ALEXANDREA. YOU WOULDN'T THINK]
39 PRINT [THAT SUCH A MAN OF THE LADIES WOULD HAVE HOMO-SEXUAL]
40 PRINT [RELATIONSHIPS . BUT IT IS RUMORED THAT THIS WAS TRUE]
41 PRINT [WITH KING NICOMEDES OF BITHYNIA.   ]
42 PRINT [ ]
43 PRINT [CAESAR WAS KNOWN TO GO ON TO FIGHT MANY GREAT FAMOUS]
44 PRINT [BATTLES IN GAUL WHERE HIS LOYALTY TO HIS TROOPS AND]
45 PRINT [PRISOERS OF WAR WAS GREAT. HE FOUGHT BATTLES IN BRITAN]
46 PRINT [BUT THIS HIROIC CAREER ENDED BY MURDER AND NOTHING BEST]
47 PRINT [DISCRIBES THIS THAN THE TWO ITEMS I HAVE HERE: ]
48 PRINT [ ]
49 PRINT [ ]
50 PRINT [IF THE GREAT CAESAR HAD NOT DIED HE WOULD HE MIGHT HAVE]
>51 PR [LASTED 10 OR 15 YEARS DUE TO A FEW EPILEPTIC SEIZURES. ]
>52 PR [HE WAS A GREAT MAN AND HE DID GREAT THINGS FOR ROME AND HE]
>53 PR [WILL ALWAYS BE REMEMBERED. ]
>END
```

Figure 5.14

talked to himself cheerfully, and thought carefully about each word. We may also speculate that the wording he chose for the messages indicate a lack of self-confidence in his own abilities, combined with a strong belief that he <u>should</u> be more successful than he is.

One interesting consequence of Dennis' work on messages for the LUNER game was that he became much more interested in landing the dynaturtle on the landing pad, since this was required to activate the messages. He became quite successful at the LUNER game from this point on.

By the time the classes ended, Dennis was ready to tackle a large interactive project. He may do so if he has a further opportunity to use a computer.

As his last activity, on the last day of classes, Dennis left a message for the MIT LOGO Group, in his procedure WERTS:

```
TO WERTS
10 PRINT [WELL ITS NOW A FINAL GOODBYE TO LOGO, SO WHEN ALL THE EGGHE
ADS AT MIT LOOK AT MY DISK I WANT THEM TO KNOW I HAD A  !*$%&%$*"! * HE
LL OF A TIME]
```

6. Donald

Donald, a student who was new to the school this year, is considered to have "above average ability" by his teachers. His most recent National achievement test scores place him in the 73rd percentile as compared to National Norms. His work in the LOGO classes was characterized by a good understanding of formal approaches to problem solving, combined with some difficulty understanding details of geometry. He was especially good at naming procedures and subprocedures, using and understanding top-down planning, making use of mathematical analysis in planning his work, and understanding the function of conditionals and stop rules. At the same time, he tended to have difficulties working "experimentally" with Turtle Geometry -- often not quite sure where the TURTLE would move next.

6.1 Donald's Working Style

Throughout his work Donald was extremely receptive to suggestions from the teacher, often making use of new ideas before he fully understood them. In this way, he was able to incorporate into his way of working, strategies that would continue to prove useful, as he gradually came to understand them through use in more than one context. He seemed to have the confidence that he could make use of the teacher's suggestions effectively and that he would eventually understand them, even if the concepts were a bit hazy at first.

Donald spent most of his class time on one long-term project: making the computer draw an elaborate head, which included a beard, hair, a hat and a flower, in addition to the usual features: eyes, ears, nose and mouth. He worked for twelve class periods on this project. He began by drawing a picture of what he wanted the head to look like, and following the teacher's suggestion, wrote out a superprocedure to draw the head, using subprocedures to add each of the features. In the course of his work, Donald had to estimate distances and angles, use arc and circle procedures, use procedures that repeat, use variables to control size and angles, and especially, learn to separate a problem into parts, to make it easier to solve. In addition, he used a recursive POLY procedure with a stop rule to make a flower for his head.

Throughout his work, Donald had difficulty understanding the effect of the state of the TURTLE at any given time. He could not always predict where the next step would occur. At times it seemed as if he had difficulty seeing exactly where the TURTLE was headed. The teaching strategy employed to help Donald deal with these problems, was to help him develop tools of mathematical analysis in order to figure out the best way to aim the TURTLE, without relying totally on experimentation. He was shown how to use a kind of "grid" to help him maneuver the TURTLE around his HEAD, and how to use the total angle already turned by the TURTLE in a given situation, to decide how much farther to turn it next. In addition, he was shown how to break up even a small problem into parts. In placing a mouth on his face, for example, he was able to separate the problem into three sub-problems: the starting point for the mouth, the orientation of the TURTLE before drawing the mouth, and which size arc to use. Thus, he was helped to overcome obstacles that might have interfered with his success while learning principles of geometry, computer programming, design and planning.

6.2 An Early Example of Donald's Approach to Problem Solving

As an example of both the strengths and weaknesses associated with Donald's structured planning approach as well as of his difficulty with visual approaches to problem solving, we consider his construction of a "house" from a square and a triangle, a common LOGO task, tackled by many students at an early stage of their LOGO experience.

At first Donald attempted an exploratory approach to solving this problem. He began by drawing a triangle on the screen, making use of TRI, a state transparent equilateral triangle procedure:

Given a TRIANGLE procedure:

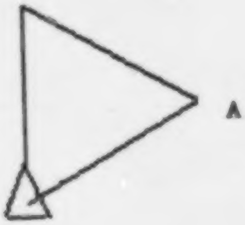


Figure 6.1

Donald spent a class period trying to add a BOX to it like this:

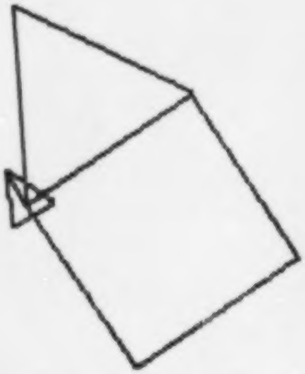


Figure 6.2

Donald's basic strategy was to try to get the TURTLE to point A of the triangle, and then use his BOX procedure. (BOX made a square by turning right.) He had difficulty figuring out how to orient the TURTLE properly. Although he came close by trial and error, he had not kept notes, and did not realize how close he was.

Donald's difficulty was due to the fact that he had established an awkward framework for the problem. He started with the initial orientation of the triangle. He then had to deal with two disorientations -- the problem of the "gap" between the TRI and BOX procedures, and the tilted orientation of the whole shape. When it was suggested that Donald make a plan by drawing a picture of the house he was trying to construct, he drew a tilted house!

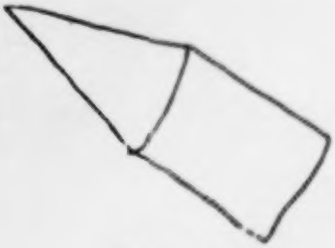


Figure 6.3

The solution that was suggested to him at the next class was to drop the plan which started with the triangle and to begin again, starting with the BOX. Once he did this, he was able to analyze the problem in a way that made it unnecessary to rotate the TURTLE to place the triangle on the BOX. He simply moved the TURTLE to the upper right hand corner of the BOX, turned it around, and used the TRI procedure, so that the first leg of the triangle was along the top of the BOX.

```
TO HOUSE
1 BOX
2 RIGHT 90
3 FORWARD 100
4 RIGHT 90
5 FORWARD 100
6 RIGHT 90
7 RIGHT 90
8 TRI
END
```

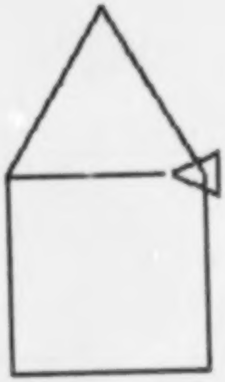


Figure 6.4

6.3 Donald's HEAD Project

Donald's major project, which lasted for four weeks and occupied twelve class periods, was to make a man's face, complete with beard, hair, hat and flower. He began by drawing the following picture in his notebook at the beginning of class 9.

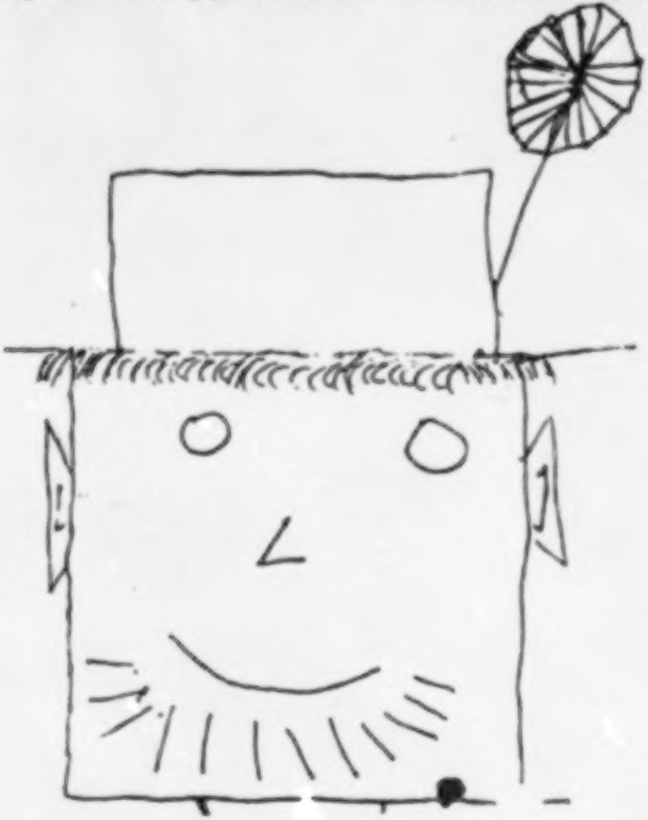


Figure 6.5

After working on the project for a few days he modified his plan somewhat, and drew this picture, which simplified the ears and the hair, and added a moustache:

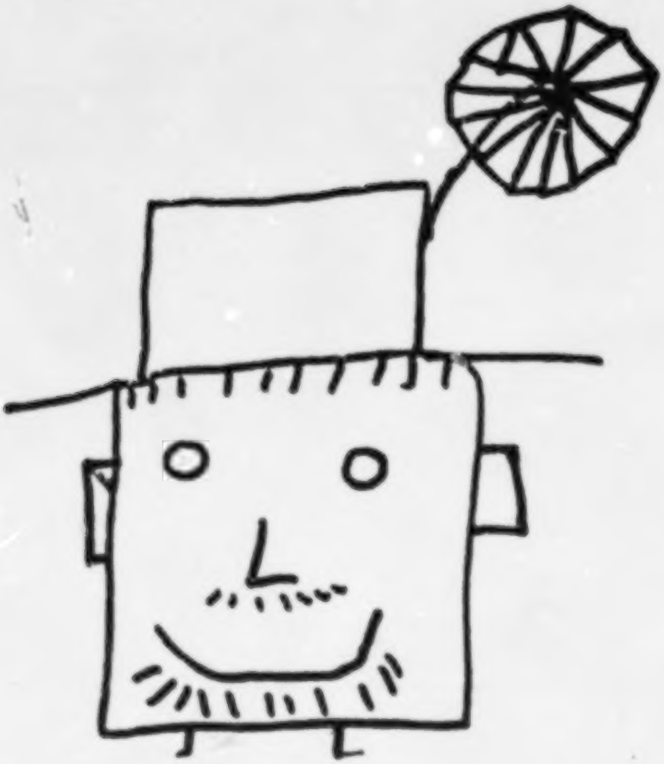


Figure 6.6

His completed project, shown in figure 6.7, looked almost exactly like his second plan (with the exception of the moustache).

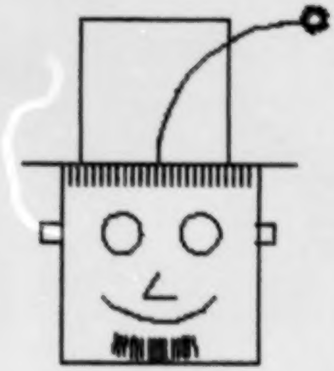


Figure 6.7

Donald began by spending one period trying to draw the entire head as a long series of direct commands. At the next class he and the teacher sat down together and developed a framework for doing the problem by breaking it into smaller parts. A superprocedure was written to serve as both an overall plan for solving the problem, and as a procedure for drawing the head. Donald included in his initial superprocedure the most important features which he intended to include in the head. Additional features, such as the ears, hat and flower, were left to be added later. Donald's original superprocedure was:

```
TO HEAD
1 BOX
2 EYES
3 NOSE
4 MOUTH
5 BEARD
6 HAIR
END
```

Using a superprocedure, and top-down planning was a major breakthrough for Donald. It meant he could concentrate on one step at a time, and not worry about having to lose or erase the picture of what had gone before. It allowed him to keep track of both his overall goal, and of exactly what he had accomplished at every stage of his work. After completing the EYES and NOSE, for example, when Donald gave the command HEAD, the computer would draw the head, in its incomplete state (see figure 6.10) and print the following message:

```
YOU HAVEN'T TOLD ME HOW TO MOUTH
AT LEVEL 1 LINE 4 IN HEAD
```

Donald's problem of building the head was thus reduced to a series of separate problems, which could each be solved in turn. For each feature, EYES, NOSE, MOUTH, etc., Donald had to figure out where to locate the TURTLE to begin drawing the feature, and the steps necessary to draw the feature itself. In some cases (MOUTH, BEARD, and HAT, for example), these two aspects of the problem were closely interrelated. In others (EYES, NOSE, HAIR, EARS) the location of the feature was relatively independent from the steps necessary to draw it.

To show how Donald solved the problem of building a head, we will first present his overall solution, including his superprocedure and each of his sub-procedures. The procedures are annotated to show exactly how the solution was achieved. Donald's work in developing a few of his key subprocedures, MOUTH, BEARD and HAT, will be analyzed in detail, to give a sense of how Donald worked on specific problems in a step by step fashion.

Here are Donald's annotated HEAD procedures:

```
TO HEAD
1 BOX
2 EYES
3 NOSE
4 MOUTH
5 BEARD
6 HAIR
70 EARS
80 HAT
85 FLOWER
END
```

(HEAD is the superprocedure) The subprocedure, BOX, draws the outside of the head. The purposes of the other subprocedures are clearly indicated by their names. Donald originally wrote the procedure up to line 6. Lines 70, 80 and 85 were added later.

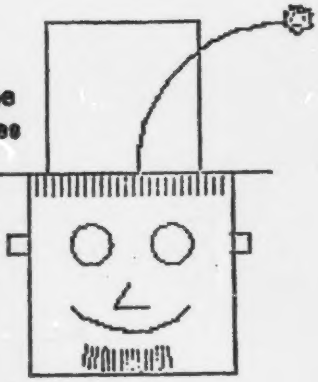


Figure 6.7

```
TO BOX
1 FORWARD 100
2 RIGHT 90
3 FORWARD 100
4 RIGHT 90
5 FORWARD 100
6 RIGHT 90
7 FORWARD 100
END
```

BOX draws a square with sides of 100.

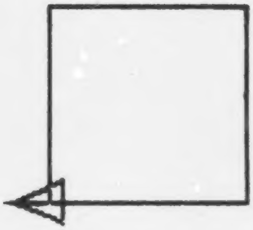


Figure 6.8

```
TO EYES                   (EYES begins where BOX leaves off)
1 RIGHT 90                moves TURTLE to face up
2 FORWARD 75              moves upward along the box
3 RIGHT 90
4 PENUP
5 FORWARD 30              moves over to draw first eye
6 PENDOWN
7 RCIRCLE 10              draws first eye
8 PENUP
9 FORWARD 40              moves over to draw second eye
10 PENDOWN
11 RCIRCLE 10             draws second eye
END
```

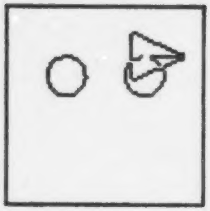


Figure 6.9

```
TO NOSE                   (begins where BOX ends)
1 PENUP
2 BACK 20                 moves over and down to position
3 RIGHT 90                the TURTLE to draw the nose
4 FORWARD 30
5 PENDOWN
6 RIGHT 30
7 FORWARD 15              draws the nose
8 LEFT 120
9 FORWARD 15
END
```

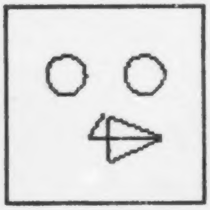


Figure 6.10

```
TO MOUTH                  (MOUTH begins where NOSE ends)
1 PENUP
2 FORWARD 20              moves over to starting point
3 RIGHT 135               sets starting angle for mouth
4 PENDOWN
5 RARC 40                 draws arc for mouth
END
```

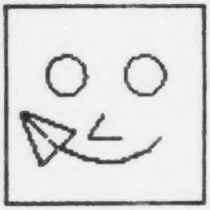


Figure 6.11

```
TO BEARD                        (BEARD begins where MOUTH ends)
10 PENUP
20 RIGHT 45                     rotates TURTLE straight up
30 FORWARD 69                   moves to top of head
40 RIGHT 90                     moves to center of head
50 FORWARD 20
60 RIGHT 90                     orients TURTLE to start
70 RIGHT 15                     drawing beard
80 REPEAT [STRING] 15
END                             draws beard by repeating STRING 15 times

TO STRING                       (STRING is a subprocedure of BEARD)
1 PENUP
2 FORWARD 80                    moves TURTLE forward with pen up
3 PENDOWN
4 FORWARD 10                    moves TURTLE forward with pen down
5 PENUP
6 BACK 90                       moves back total distance with pen up
7 LEFT 2                        turns TURTLE LEFT 2 to set up next
END                             STRING
```

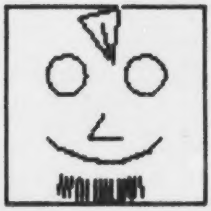


Figure 6.12

```
TO HAIR                         (HAIR begins where BEARD ends)
10 RIGHT 15
20 BACK 9
30 RIGHT 90                     reorients TURTLE, and moves over
40 FORWARD 50                   to upper right hand corner of
50 LEFT 90                      the head to draw the hair
60 REPEAT [HAIRY] 25
END                             draws hair by repeating HAIRY
                                25 times
```

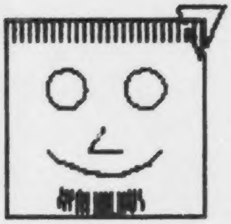


Figure 6.13

```
TO HAIRY                    (HAIRY is a subprocedure of HAIR)
5 PENDOWN
10 FORWARD 10               moves forward and back to draw a hair
20 PENUP
30 BACK 10
40 LEFT 90
50 FORWARD 4                moves over and reorients TURTLE to draw
60 RIGHT 90                 the next hair
END
```

```
TO EARS                     (EARS starts where HAIR ends)
10 FORWARD 30
20 LEFT 90                  moves down the side of the head
30 PENDOWN                  to draw the first ear
40 EAR                      subprocedure EAR draws the first ear
50 PENUP
60 FORWARD 100              moves the TURTLE across the head
70 PENDOWN
80 EAR                      subprocedure EAR draws second ear
END
```

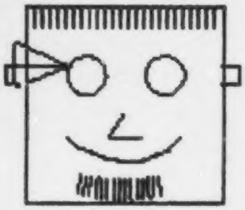


Figure 6.14

```
TO EAR                      (EAR is a subprocedure of EARS)
10 FORWARD 10
20 RIGHT 90
30 FORWARD 10
40 RIGHT 90                 draws a three-sided box to make an "ear"
50 FORWARD 10
END
```

```
TO HAT                      (HAT begins where EARS ends)
10 LEFT 90
20 FORWARD 30               moves TURTLE to top of head and
30 LEFT 90                  over to right hand corner
40 FORWARD 20
50 BACK 140
60 FORWARD 35               moves forward to start hat
70 LBOX 75                  subprocedure LBOX :SIZE draws
79                          a box with sides of 75
END
```

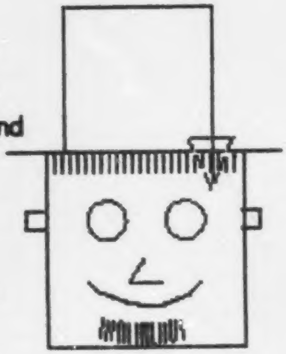


Figure 6.15

```
TO LBOX :SIZE                  (LBOX is used as a subprocedure
1 FORWARD :SIZE                of HAT)
2 RIGHT 90
3 FORWARD :SIZE
4 RIGHT 90                     LBOX draws a box of variable size
5 FORWARD :SIZE
6 RIGHT 90
7 FORWARD :SIZE
END

TO FLOWER                      (FLOWER starts where HAT ends)
10 RIGHT 90
20 FORWARD 35                  lmoves over and orients TURTLE
30 RIGHT 90                    to start drawing the flower
40 RARC 75                     draws the stem of the flower
50 LEFT 90                     centers the flower on the stem
60 BACK 5
70 POLY 10 100                 draws the actual "flower"
END
```

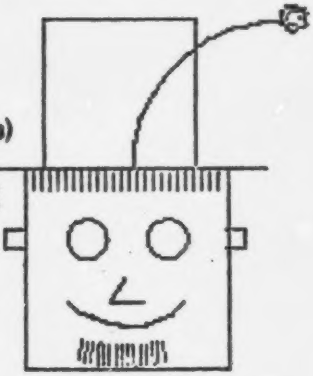


Figure 6.7

In developing each of the subprocedures outlined above, Donald had to solve two related problems: how to use TURTLE commands to draw the feature, and where to place the TURTLE before starting to draw the feature. As he worked, Donald created each feature in the order listed in the superprocedure, HEAD. The subprocedure for each new feature had to begin where the previous one left off. Each new feature presented a new challenge, and since Donald had difficulty solving problems by experimentation alone, he needed to develop analytical skills in order to divide each of his subprocedures into its sub-parts, and to solve each sub part. Donald's work in drawing three key features, the mouth, beard and hat, will now be discussed in order to illustrate the problems he encountered and the analytical skills he developed to resolve them.

Donald decided to use an arc procedure to draw the mouth on his head (see figure 6.II). In order to place the mouth on the face in a symmetrical position, Donald had to coordinate four different aspects of the problem:

-- where to place the TURTLE in order to draw the mouth
-- at which angle to orient the TURTLE to draw the mouth
-- the size of the input to be used with the arc procedure
-- whether to use a right or left turning arc procedure.

Donald spent most of one period experimenting unsuccessfully, varying these elements in an uncoordinated way. The teacher then suggested working separately on the different aspects of the problem to facilitate a solution. It was also suggested that Donald clear the screen and work on the mouth by itself, without the rest of the face, to simplify the process.

At the beginning of the next period, Donald sat down with the teacher for a "lesson" on the geometry of quarter arcs. Since Donald knew that four arcs made a complete circle and that the TURTLE turned through 360 degrees in completing a circle, he realized through discussion, that each arc procedure turned the TURTLE 90 degrees. Using this fact, and taking note of the horizontal position of the TURTLE after finishing drawing the nose (figure 6.10), he was able to figure out that the TURTLE had to be turned through an angle of 135 degrees to orient it so that it would draw a symmetrical curve.

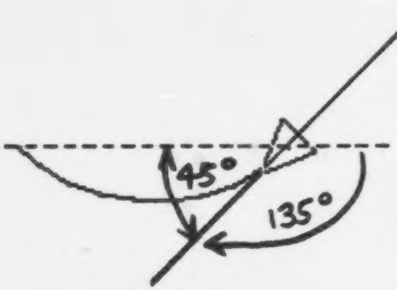


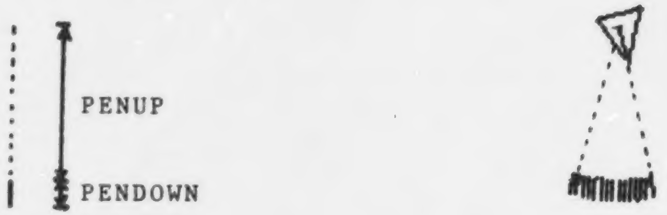
Figure 6.16

Once he understood how to orient the TURTLE, it was easy for Donald to determine a size for the mouth that corresponded with the other features of his head, and to locate the starting point for the mouth by trial and error. He did this by moving the TURTLE over from where the nose ended, until the arc was positioned symmetrically in the center of the head (see figure 6.11).

In order to draw a beard on his head, Donald also had to separate his problem into three different aspects:

- --how to draw the beard
- --where to locate and orient the TURTLE to draw the beard
- --how to move the TURTLE to that point, from the point where the MOUTH subprocedure ended.

Here, the teacher provided a suggestion for drawing the beard, which Donald accepted. The beard was to be drawn as a series of "lines", all emanating from a single point. As the TURTLE moved, it kept its pen up for most of the line, putting the pen down only to draw the "hair" of the beard (Figure 6.17).



STRING

REPEAT [STRING] 15

Figure 6.17

In writing his procedure STRING, Donald decided how long the PENUP and PENDOWN portions of the line would be, as well as how much to rotate the TURTLE each time the line was repeated. Once STRING was defined, he had to figure out how many repeats he wanted, as well as how to orient the TURTLE to make the beard symmetrical. After a few tries he figured out that if he turned the TURTLE RIGHT 15 (from a straight down position) and repeated STRING 15 times, he would get a symmetrical beard (See figure 6.17).

Now Donald had to center the TURTLE at the top of the head, in order to draw the beard on the face. (The TURTLE was oriented at an angle of 45 degrees to the horizontal after drawing the mouth. See figure 6.11). At first Donald tried to orient it vertically by trial and error -- but had difficulty finding the correct angle. A short geometry lesson helped him realize that the TURTLE needed an additional rotation of 45 degrees to be heading in the proper direction. From that point he was able to easily center the circle on the top of his head. He found that he still needed to move it a distance of ten TURTLE steps below the top of the head, in order to locate the beard properly below the mouth. The steps required to accomplish this are given in his procedure BEARD

In placing a hat on the head, Donald once again had to coordinate several related factors:

--the size of the box needed to draw the hat itself

--the overall width of the brim

--the distance the TURTLE had to move in from the edge of the brim before drawing the box, in order to make it symmetrical.

This time Donald worked out the problem on his own, using a variable box procedure, LBOX :SIZE, that he had developed at an earlier stage of his work. Once again, Donald had difficulty coordinating all the variables. This time, he invented a kind of "coordinate system" to help him keep track of what he was doing: after each try he counted the "hairs" on both sides of his hat to see if it was centered. Although his final result was not exactly symmetrical he was able to obtain a satisfactory solution using this approach (see figure 6.15).

6.4 Conclusion

In summarizing Donald's work we note that his was the longest and most involved Turtle Geometry project undertaken by any of the children in our trial classes. By using the top-down mode of operation, he was able to understand both the overall goal, and where he was in the process at any particular moment. Thus he was able to meet each challenge as just one small problem to be overcome, so that he did not become discouraged about the whole project.

In the course of his work Donald encountered estimation of distances and angles, the geometry of arcs and circles, the total turtle trip theorem, and the use of both grid-based and intrinsic coordinate systems. He learned to use subprocedures and sub-subprocedures, to use patterned procedures making use of a REPEAT command, to make use of variables to control the size and shape of his "hat" and "flower" and to use a POLY procedure with a conditional stop rule. Although Donald only "learned" these approaches to the extent necessary to solve the particular problems inherent in his project, each succeeding use of the same concept, reinforced his exposure to it, deepening his sense of mastery.

Perhaps the most striking thing about Donald's work was that he was not usually able to solve his problems by experimentation and visual examination of the results. From his first HOUSE procedure, right through the HEAD, all the way to the FLOWER, he made use of analysis, combined with experimentation to solve his problems. He often needed help with the analysis, but he understood the analytical techniques he was shown and he was able to make use of them. Scattered throughout his notebook are little drawings on graph paper, either by Donald or the teacher, showing how he analyzed his work. It was his regular use of drawings like these that distinguished Donald's work from that of his classmates. A few of them are presented here in figure 6.18.

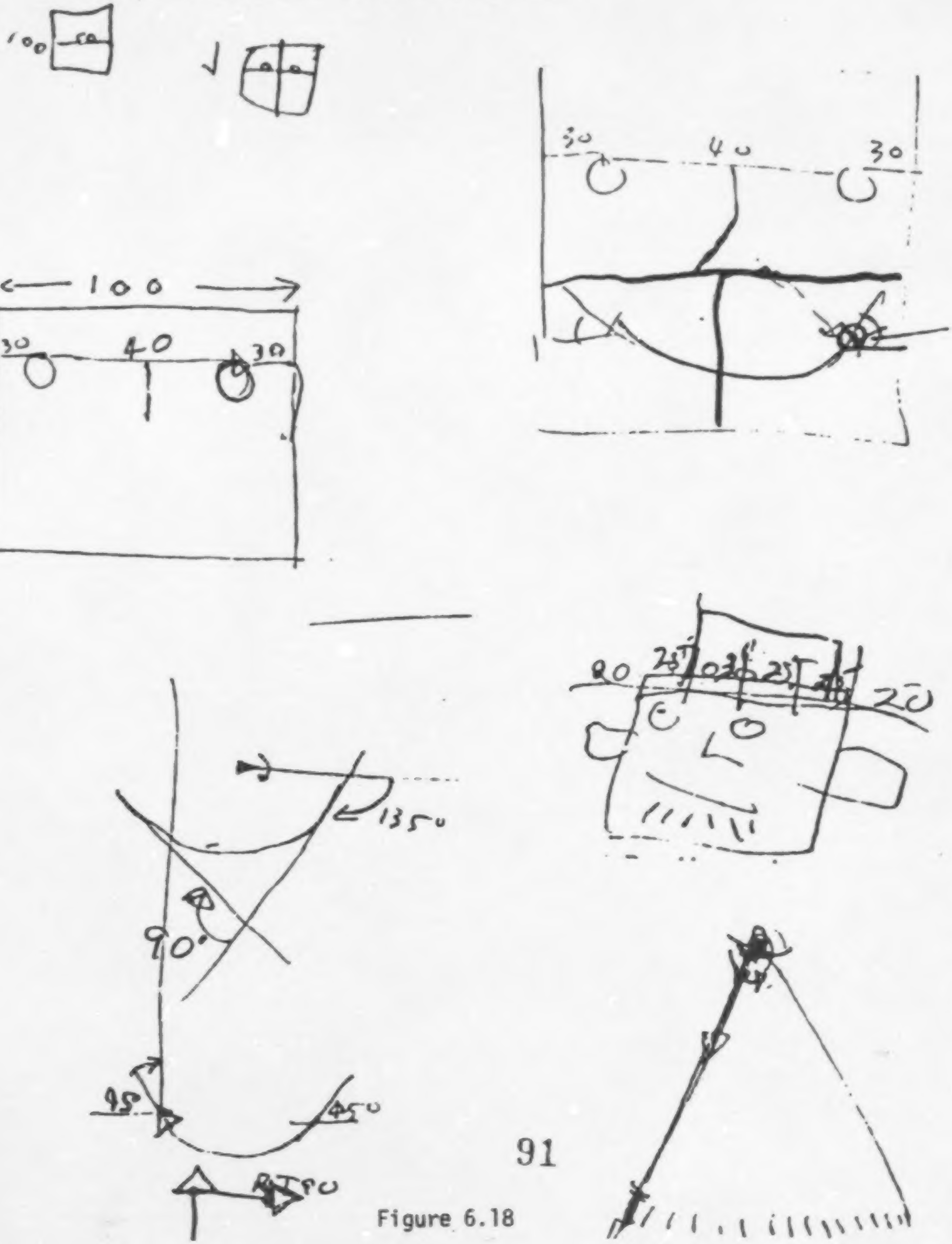

100
50
100
30
40
30
30
40
30
135°
90°
45
45°
RT 90
20
25
20

Figure 6.18

7. Gary

Gary is considered to be "extremely bright" by his teachers. On his sixth grade school achievement tests his overall score placed him in the 83rd percentile. His teachers report that they find it difficult to find ways to challenge him within his regular school program, while at the same time reporting "peculiar gaps" in his academic knowledge -- in the area of standard arithmetical skills, for example.

Gary found LOGO to be an exciting challenge. He worked on four major projects each of which led to significant new learning in different areas: using arcs and circles to draw a face; creating a simple math quiz; drawing and animating a starship; and writing a computer program capable of "understanding" Morse Code. During the course of his work, Gary mastered the use of recursion and variables in a number of different contexts; he understood the use of conditionals and "branching"; he learned to write state transparent procedures, and to use superprocedures with modular subprocedures. In his last project he was beginning to make use of recursion in list and word processing, and learned the concepts of "empty list" and "empty word".

The teaching strategy used with Gary was to offer him simple models of a particular kind of procedure, give him the information he needed, and leave him alone to elaborate on the model, providing help only when asked. When one phase of a project was finished, additional challenges were suggested occasionally he was asked to alter or improve his work. In this way, Gary was able to move ahead on his own, at as fast a rate as he wanted.

1. Gary's Working Style

Of all our experimental subjects, Gary, who had had some prior exposure to computers, seemed the most predisposed to success in LOGO. He combined a strong interest in computers with a learning style that encompassed both analytical and trial-and-error approaches. He absorbed new ideas voraciously, and rarely had to be shown something twice. He tackled extremely ambitious projects, and always stayed with a project until some kind of completion was achieved.

Gary's work demonstrated some clear characteristics that set him apart from most of the other students: Gary easily understood the use of a procedure as an entity, recognizing the usefulness of naming a series of steps, and thereafter considering them as a "unit"; he often wrote procedures without trying out the steps individually first, recognizing that the series of steps could be considered to have a "total effect," as though it were a single command.

Gary had faith in his ability to solve problems by reasoning as well as trial-and-error. He was constantly trying things out "in his head", making use of a number of "abstract principles" to simplify and debug his work as he went along. A series of FORWARD and BACK commands would easily be combined into one command. Left/right reversibility would be used to correct an error. At one point, for example, Gary had typed RIGHT 99, and realized from the effect that he should have used LEFT 99. He then used the computer to add 99 + 99, and typed LEFT 198. In writing his procedure later, he simply used the correct command LEFT 99, without ever having tried it explicitly.

Gary tended to "plunge into a problem" impetuously, with very little advance planning, drawing on a quick analysis, based on partially understood ideas. He then enjoyed the process of debugging his original idea, or of moving in a new direction, if his result was significantly different from what he had intended. In the first class, Gary noticed that repeating a simple three step procedure made a "pattern" that looked something like a "circle". When another student suggested making a smaller circle inside the first one, Gary started a small investigation of curvature. He began to make a new circle by using the same procedure with smaller inputs. When his new circle came out larger than the original one, Gary was surprised and delighted. He then tested another approach -- make all the inputs exactly half of the original, which led to a "circle" of almost the same size as his first one. And so on, until he had tried many variations.

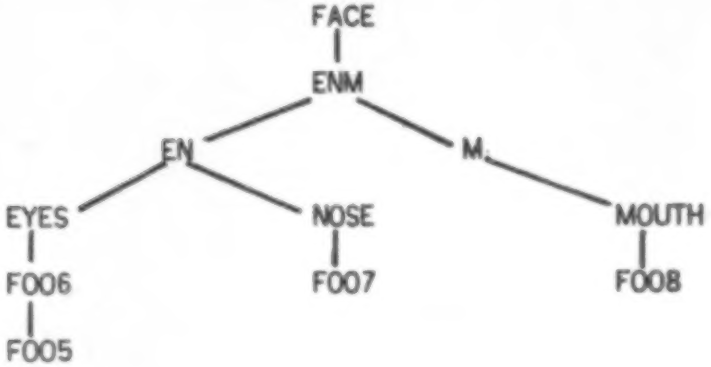
Gary usually worked in a step-by-step fashion, rather than plan his work in advance. While he generally had an overall idea of what he was trying to do, he tended to incorporate subprocedures one after another, rather than to break down his problem into parts and plan his subprocedures in advance. He showed that he was capable of using a more structured approach. When he was asked by his teacher to rewrite his STARSHIP procedure, he created a set of simple modular subprocedures to draw the starship.

Gary often sought out bugs, testing for extreme situations: the largest possible inputs, the largest number of REPEATs, situations which would produce error messages, as a way of understanding both the capabilities and limitations of the computer, beyond the needs of any specific project on which he was working.

Gary's work was goal-directed and ambitious. During his seven weeks of LOGO classes he worked on four major projects. While he enjoyed brief detours, such as the "circle" exploration described above, his work was usually directed quite specifically at his particular immediate goal. Between projects, he often appeared to be restless; once a new task was selected, he was off and running again.

2. Gary's FACE Project

Gary's first project was to draw a complicated "face" built from a large number of subprocedures. The project involved learning to use superprocedures, subprocedures, and a great deal of Turtle Geometry. Gary used functional procedure names, abbreviated procedure names, and "nonsense" names, all in a rather elaborate scheme to "hide" the sub-procedures which actually "did the job". This complex hierarchy of subprocedures was extremely difficult for Gary himself to debug, and he often had to trace through the entire "tree structure" of his project to find a bug in a particular procedure.



In carrying out his FACE project Gary quickly developed a facility with Turtle Geometry: basic TURTLE commands; RIGHT/LEFT reversibility; mastery of 90 degree angles; the geometry of arcs and circles; and the use of an implied axis of symmetry. Although his design was symmetrical, it was not constructed by simply reversing RIGHT/LEFT commands to make it symmetrical about its center. Gary's more complex method involved working "from the outside, in." To draw the eyes, for instance, Gary moved the TURTLE to the outside of the right-hand circle and drew the circle. Then he calculated the distance of two diameters, moved the TURTLE to the outside of the left-hand circle, and drew that circle. (See Figure 7.1, and Gary's procedure, F006).

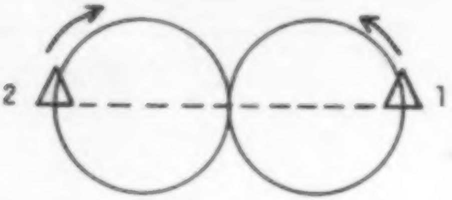


Figure 7.1

```
TO FACE
10 ENM
END
```

```
TO ENM
10 EN
20 M
END
```

```
TO EN
1 EYES
2 NOSE
END
```

```
TO M
1 MOUTH
END
```

```
TO NOSE
1 F007
END
END
```

```
TO EYES
1 F006
END
```

```
TO MOUTH
10 F008
END
```

```
TO F006
1 F005
2 PENUP
3 LEFT 90
4 FORWARD 80
5 PENDOWN
6 RIGHT 90
7 RCIRCLE 45
8 PENUP
9 RIGHT 90
10 FORWARD 160
11 LEFT 90
12 PENDOWN
13 LCIRCLE 45
14 HIDETURTLE
END
```

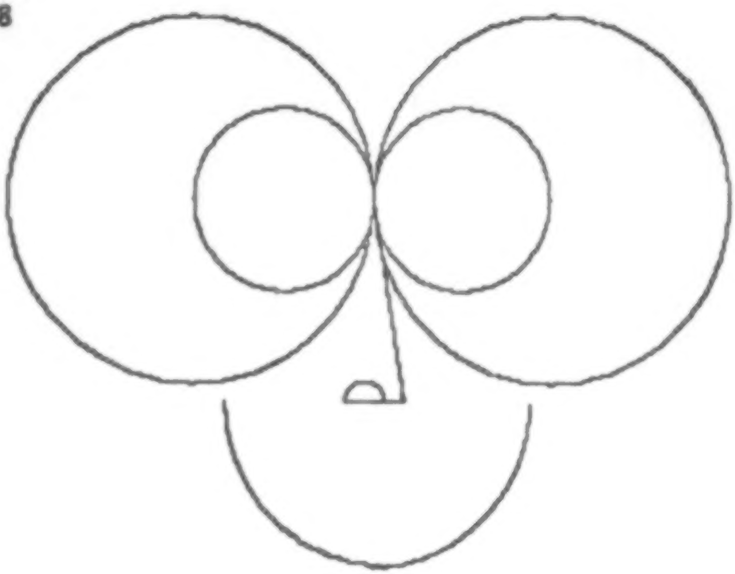


Figure 7.2

```
TO FOO5
1 LCIRCLE 90
2 RCIRCLE 90
END

TO FOO8
1 PENUP
10 PENUP
20 FORWARD 70
30 PENDOWN
40 RIGHT 90
50 PENUP
55 RIGHT 90
60 FORWARD 166
70 RIGHT 90
80 FORWARD 70
90 LEFT 90
100 PENDOWN
110 LARC 80
120 HIDETURTLE
END
```

```
TO FOO7
1 PENDOWN
2 S
3 FORWARD 100
4 RIGHT 99
5 FORWARD 30
6 RIGHT 90
7 RARC 10
8 RARC 10
9 HIDETURTLE
END

TO S
6
10 PENUP
20 SHOWTURTLE
30 LEFT 90
40 FORWARD 80
50 LEFT 90
60 LEFT 9
70 PENDOWN
END
```

3. Gary's Math Quiz

Gary's second project involved the use of conditionals, PRINT statements, the naming of variables, and random numbers to create a "math quiz" which gave a user a series of two-digit addition problems. Although he planned to extend the project to include subtraction, multiplication and division, he decided to go on to other activities after completing the addition portion of the quiz. During the following year, Gary went back to this project and completed it, making use of LOGO computer provided by his school system.

While working on the math quiz Gary encountered an interesting bug which was typical of the kinds of problems that often resulted from Gary's working style. Gary would plunge into a problem "headfirst", with very little planning or consideration for the effects of an action that might go beyond his focus of the moment. In making his math quiz, he originally had a set of procedures named MATH, MATH1, MATH3. He decided that they were ordered incorrectly, and so decided to EDIT the titles of the procedures, so that the superprocedure would be called MATH, the first subprocedure would be MATH1, the second, MATH2, etc. What he forgot while doing this, was that he would also have to change all the procedures themselves, so that they would be calling the correct subprocedures. When he finished changing the names around, all his procedures suddenly stopped working. He was able to debug this situation himself by printing out his procedures and "playing computer".

```
TO MATH
10 PRINT [WOULD YOU LIKE TO HAVE A MATH TEST?]
15 MAKE "ANS REQUEST
20 IF :ANS = [YES] PRINT [WELCOME TO THE WORLD OF MATH!] MATH1 STOP
30 IF :ANS = [NO] PRINT [O.K. COME BACK AGAIN!] STOP
END

TO MATH1
5 MAKE "NUM1 WORD RANDOM RANDOM
6 IF FIRST :NUM1 = 0 GO 5
7 MAKE "NUM2 WORD RANDOM RANDOM
8 IF FIRST :NUM2 = 0 GO 7
10 PRINT ( SENTENCE [%%] :NUM1 )
15 PRINT [+]
20 PRINT ( SENTENCE [%%] :NUM2 )
21 PRINT [______________]
25 MAKE "ANS TYPEIN
30 TST :ANS = :NUM1 + :NUM2
40 IFTRUE PRINT [CORRECT!] MATH2 STOP
50 IFFALSE PRINT [TRY AGAIN!]
60 GO 10
END

TO MATH2
10 PRINT [WOULD YOU LIKE TO HAVE ANOTHER PROBLEM?]
20 MAKE "ANS REQUEST
30 IF :ANS = [YES] PRINT [O.K. HERE WE GO AGAIN!] MATH1 STOP
40 IF :ANS = [NO] PRINT [ALL RIGHT SEE YOU NEXT TIME!] STOP
END

TO MATH3
5 PRINT [WELCOME TO THE WORLD OF MATH!]
10 PRINT [17 + 28=]
20 MAKE "ANS TYPEIN
30 TEST :ANS = 17 + 28
40 IFTRUE PRINT [CORRECT!] STOP
50 IFFALSE PRINT [TRY AGAIN.]
60 GO 10
END
```

4. Gary's Starship Project

Gary's third major project was the drawing and animation of a starship (Figure 7.3). At first he constructed a long, involved step-by-step procedure, which required some involved, rather frustrating debugging. Gary's debugging resulted in the addition of even more lines to his one basic procedure.

Although Gary successfully debugged the procedure, his teacher suggested that he redo the entire project, making use of simple procedures and subprocedures. This time, having experienced both extremes -- an unnecessarily complex hierarchy of subprocedures in his FACE project, and an unnecessarily long single procedure for his STARSHIP, Gary constructed a set of modular subprocedures to draw the starship:

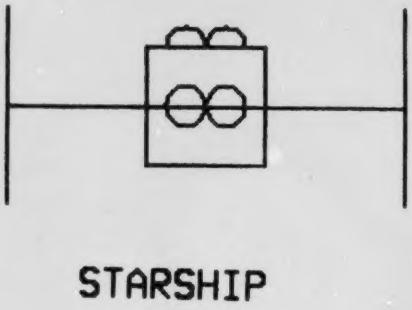


Figure 7.3

Old Starship Procedure

```
TO STARSHIP
10 RIGHT 90
20 FORWARD 100
30 LEFT 90
40 FORWARD 50
50 RIGHT 180
60 FORWARD 100
70 PENUP
71 LEFT 180
72 FORWARD 50
73 LEFT 90
74 FORWARD 100
75 RIGHT 90
90 LEFT 90
95 PENDOWN
100 FORWARD 100
105 RIGHT 90
110 FORWARD 50
120 LEFT 180
```

(continued, next page)

New Starship Procedures

```
TO STARSHIP
10 STA
20 WINGR
30 WINGL
END
```

```
TO STA
5 WRAP
10 C
20 LI 100
END
```

```
TO WINGR
10 MO
20 RIGHT 90
30 LI 50
40 MOVE
END
```

(Old Starship, cont.)

```
130 FORWARD 100
140 PENUP
141 RIGHT 180
142 FORWARD 50
143 RIGHT 90
144 FORWARD 100
145 LEFT 90
155 PENDOWN
160 RIGHT 90
170 FORWARD 30
180 LEFT 90
190 FORWARD 30
200 LEFT 90
210 FORWARD 60
220 LEFT 90
230 FORWARD 60
240 LEFT 90
250 FORWARD 60
260 LEFT 90
270 FORWARD 30
280 PENUP
290 LEFT 90
291 FORWARD 30
292 RIGHT 90
300 HIDETURTLE
310 PENDOWN
320 RCIRCLE 10
330 LCIRCLE 10
340 PENUP FORWARD 30
345 PENDOWN
350 RARC 10
360 RARC 10
370 PENUP RARC 10
380 RARC 10
390 LARC 10
400 LARC 10
410 HIDETURTLE
END
```

(New Starship, cont.)

```
TO WINGL
10 MOV
20 LEFT 90
30 LI 50
35 RIGHT 90
40 HIDETURTLE
END

TO C
10 SQ.1
20 PENDOWN RCIRCLE 10
30 LCIRCLE 10
40 PENUP FORWARD 30
45 PENDOWN
50 REPEAT [RARC 10] 2
60 PENUP REPEAT [RARC 10] 2
70 PENDOWN REPEAT [LARC 10] 2
80 PENUP REPEAT [LARC 10] 2
90 BACK 30
END
```

```
TO MO
10 RIGHT 90
20 FORWARD 100
30 LEFT 90
END

TO SQ.1
5 PENUP
10 RIGHT 90
20 FORWARD 30
30 RIGHT 90
35 PENDOWN
40 FORWARD 30
50 RIGHT 90
60 FORWARD 60
70 RIGHT 90
80 FORWARD 60
90 RIGHT 90
100 FORWARD 60
110 RIGHT 90
120 FORWARD 30
130 RIGHT 90
135 PENUP
140 FORWARD 30
150 FORWARD 30
END
```

```
TO MOV
10 LEFT 90
20 FORWARD 100
30 RIGHT 90
END

TO LI :LE
5 PENDOWN
10 RIGHT 90
20 FORWARD :LE
30 LEFT 180
40 FORWARD 2 * :LE
50 RIGHT 180
60 FORWARD :LE
70 LEFT 90
END
```

5. Gary's Morse Code Project

For his last project, Gary decided to create a Morse Code Translator as the first step of a project to actually transmit morse code over radio waves, which he had read about in a computer hobbyist magazine. In creating the Morse Code translator he made use of FIRST and BUTFIRST comands in connection with list and word processing; his recursive procedures used the concepts of "empty word" and "empty list" in their STOP rules; and he used conditional statements to decide which particular set of Morse Code symbols to output.

The morse code translator was built up by first creating a lengthy procedure, CODE, which output the correct sequence of dots and dashes for any letter or number:

```
TO CODE :LETTER
10 IF :LETTER = "A OUTPUT ".-
20 IF :LETTER = "B OUTPUT "-...
30...
END
```

The procedure PRI: WORD prints the correct sequence of letters for an entire word:

```
TO PRI :WORD
10 IF :WORD = " STOP
20 TYPE CODE FIRST :WORD
30 TYPE "/
40 PRI BUTFIRST :WORD
END

PRI "HELLO

  .../././.-../.-../---/
```

The procedure PR12 :SENT, prints the correct sequence of letters for an entire sentence:

```
TO PRI2 :SENT
10 IF :SENT = [ ] STOP
20 PRI FIRST :SENTENCE
30 TYPE "
40 PRI2 BUTFIRST :SENT
END

PRI2 [HELLO HOW ARE YOU]

  .../.-../.-../.../---/.--/.-/.-././-.---/---/..-/
```

In creating these procedures Gary had to understand the difference between words and lists, and how this effected the wording of the stop rules in PRI (which manipulated words) and in PRI2 (which manipulated lists of words). He also had to carry out a process of trial and error to determine the location of the stop rule in each procedure. When the series of LOGO classes ended, Gary was engaged in the process of reversing the code -- that is, writing a set of procedures which would take a string of Morse Code symbols as input, and print out an English sentence.

6 Conclusions

Gary absorbed a great deal in approximately 25 hours of LOGO classes. His projects involved a number of different content areas: Turtle Geometry, interactive programming, animation, list processing, etc. His enthusiasm remained at a fever pitch throughout the series of classes. Gary actively resisted leaving class each day and when the cycle of LOGO classes was finished, Gary helped establish an after school "computer club," so that he could carry on his work.

The computer activities provided the kind of challenge and the scope of intellectual activities that Gary wanted and needed to develop his abilities most fully. The fact that this challenge and scope was not always present his regular classes was attested to frequently by his classroom teachers. A LOGO capability in a classroom could help teachers meet the needs of students like Gary.

8. Harriet

Harriet is one of the brightest, most academically successful students in our experimental sample (her national achievement test scores placed her in the 99th percentile). In her regular school work, Harriet loves reading and writing, but finds mathematics and science "boring". She enjoys "playing with words," and prides herself on the uniqueness of her ideas.

While Harriet was one of the most able LOGO students, she was one of the least enthusiastic. She carried out two interesting and unique Turtle Geometry projects, but announced that she was "bored with drawing pictures". She rarely initiated project ideas, but responded very well to suggestions. She carried out two complex advanced projects: programming the computer to play "tictactoe", and writing a program that would generate a series of madlib stories, by randomly substituting nouns, verbs, adjectives, etc. for those in the original story.

Harriet seemed to like the idea of each of her large projects, and was very quick to learn the LOGO commands and programming ideas necessary to carry them out. Although she found the necessary detailed work of planning, typing, testing and debugging to be quite tedius, she worked quite steadily over a number of class periods, and finished both projects quite satisfactorily.

Harriet was extremely responsive to the teacher's sugestions. She could often understand and absorb a new idea at the first hearing -- resolving minor problems by experimentation. Harriet was also responsive to error messages from the computer, corrected her mistakes when she could, and was conscientious about asking for help when she needed it. Harriet enjoyed helping others and developed a strong relationship with Tina, whom she regularly helped with details of her LOGO work.

8.1 Harriet's Working Style

Harriet's way of working in the LOGO classes offered a clear demonstration of the skills and strategies that make her an extremely successful student.

Harriet chooses interesting projects and sticks with them.

She has a sense of what it means to be "creative", and has the ability to carry it off. Her ideas are always a little bit different than those of her peers, and Harriet values this. Harriet recognizes the value of completing a project, and is willing to continue with the work necessary to fill in the details of an idea, even after it is no longer "fun" or "creative". In this way, her completed work tends to have a superior quality. On the other hand, while completing a project, Harriet would often declare that the necessary work was "boring".

Harriet is willing to explore on her own.

She attends to error messages, and corrects herself when she can. When she gets stuck she immediately asks for help -- and is usually aware of exactly where the confusion is.

Harriet's dribble files revealed a lot of exploratory activities. She would try out new commands, and was able to debug spelling or syntax errors by interpreting the error messages she received. If she was confused the first time a particular bug appeared, she would usually know exactly what to do the second time.

Harriet follows directions explicitly.

She absorbs new approaches without being confused. She applies the same approach of confident exploration to new ideas she has been taught, that she applies to her own experimentation. Only when a bug appears does she question what she is doing, and ask for help. Harriet's two large projects, her TICTACTOE game, and her MADLIB story generator, both involved her with the use of new LOGO commands, new syntax and new programming ideas. She seemed to absorb most of these ideas on first exposure -- copying a formula, rather than analyzing. Occasionally this led to a bug which she did not understand. Usually it led to success in using the new idea with little difficulty.

Harriet adds her own creative variations to other people's ideas.

Harriet can easily find a new "wrinkle" or a creative way to do something. Her first project (SMILY -- see below), was a totally unique way to draw a face. She used an extremely clever set of words, to make a funny set of stories, in her madlib project. In this way, Harriet was able to adapt someone else's idea -- a madlib procedure, for example -- and make it uniquely and specially her own.

There were, however, some aspects of Harriet's work that tended to interfere with her success as a student.

Harriet did not show much initiative in choosing projects.

Like Gary, when she finished one piece of work, Harriet tended to feel aimless, at loose ends, until she had redirected her energy. At these times she seemed to want direction from another person. When offered a choice of alternatives, she was able to easily choose among them -- but the ideas themselves did not come easily to her.

Harriet's major approach seemed to be "exploratory" rather than "analytical."

She was very comfortable at the "try it out, and debug it," approach; but much less successful with planning and using analytical techniques to solve problems, and avoid difficulties. (See below for clear examples of this difficulty in Harriet's work with Turtle Geometry).

Harriet tended to get bored with the details of a project, long before it was completed.

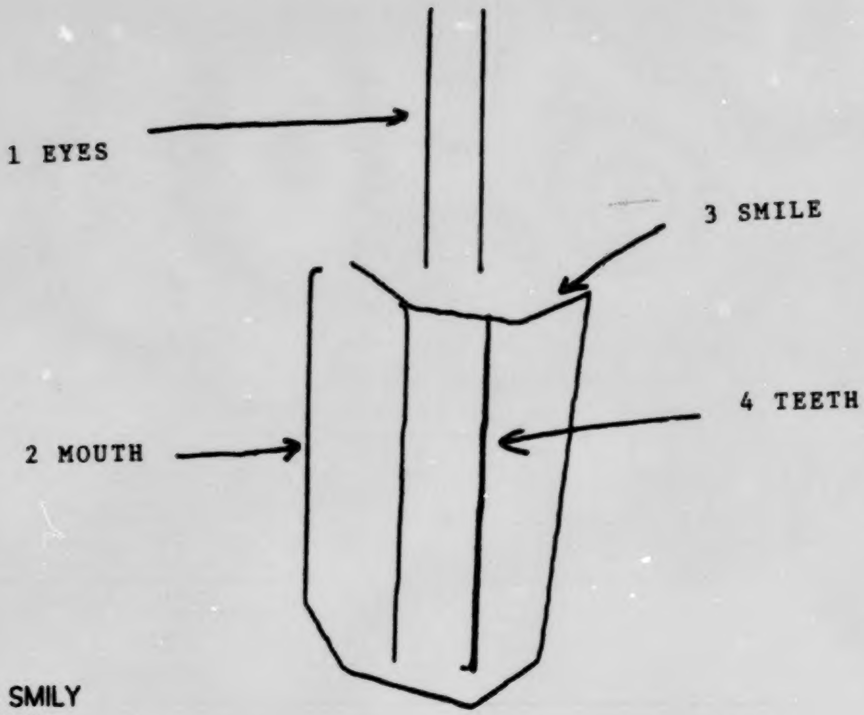
She would persevere out of a commitment to "completeness", rather than out of a sense of enjoyment of her work. This left her with a rather anti-climatic feeling at the end of a project, rather than the satisfaction of a job well done.

In part, this kind of feeling is a result of her lack of analysis -- she would start a long project, intrigued with the idea, not realizing how much detailed work was involved. Partly it was a result of the complex projects that were suggested to her, to give her a better sense of what she could accomplish with the computer. She seems to have had some difficulty finding the proper balance between the originality and power of what she wanted the computer to do -- and the complexity of detail needed to bring it off. (See a discussion of Harriet's TICTACTOE and MADLIB projects for more detailed description of this kind of dilemma.)

8.2 Harriet's Work with Turtle Geometry

Harriet started out with an excellent sense of how to manipulate the TURTLE, together with a strong desire to explore beyond what she alrady knew, to find new ways of using the computer. When she was being shown how to use FORWARD and RIGHT commands for the first time, Harriet was already asking how to leave spaces in a drawing (she was immediately shown how to use PENUP and PENDOWN) As soon as she began drawing with the TURTLE she asked about making curved lines. She quickly got into spinning designs, reversing spins, and finding the largest input she could to a SPIN command.

Harriet's very first TURTLE drawing exploration led to a project taking three class periods to complete -- her unique "smily face." (Figure 8.1)


1 EYES
3 SMILE
4 TEETH
2 MOUTH

```
TO SMILY
1 EYE
2 MOUTH
3 SMILE
4 TEETH
END
```

Figure 8.1

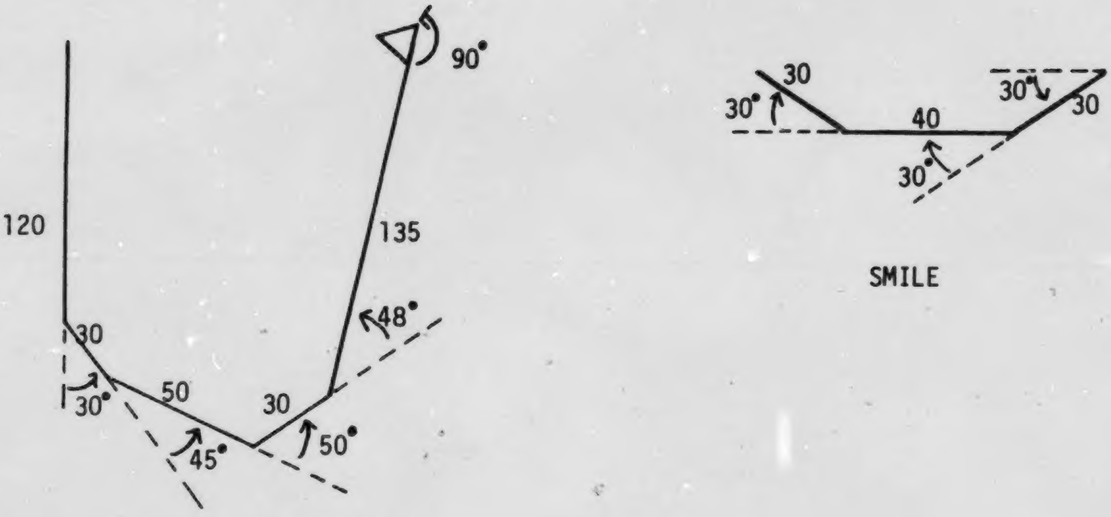
```
TO EYE
1 FORWARD 99
2 PENUP
3 LEFT 90
4 FORWARD 20
5 LEFT 90
6 PENDOWN
7 FORWARD 99
END
```

```
TO SMILE
1 LEFT 30
2 FORWARD 30
3 RIGHT 30
4 FORWARD 30
5 FORWARD 10
6 RIGHT 30
7 FORWARD 30
END
```

```
TO MOUTH
1 PENUP
2 RIGHT 90
3 FORWARD 43
4 LEFT 90
5 PENDOWN
6 FORWARD 120
7 LEFT 30
8 FORWARD 30
9 LEFT 45
10 FORWARD 50
11 LEFT 50
12 FORWARD 30
13 LEFT 60
14 RIGHT 12
15 FORWARD 135
16 LEFT 90
END
```

```
TO TEETH
1 LEFT 180
2 PENUP
3 FORWARD 30
4 RIGHT 55
5 PENDOWN
6 FORWARD 130
7 PENUP
8 LEFT 90
9 FORWARD 30
10 LEFT 90
11 PENDOWN
12 FORWARD 130
13 LEFT 180
14 FORWARD 130
END
```

As Harriet worked on this project, her approach was to explore each part of the figure, writing down the correct steps -- then make each part a procedure. The greatest difficulty was in making the mouth. Although Harriet had a very good sense of what she wanted to do, and a plan drawn on paper, she had difficulty making the mouth symmetrical. She worked by a process of trial and error, and did not do any calculations as she worked. Once she finished the mouth, she made use of symmetry to make the SMILE, the upper part of the mouth.



MOUTH

SMILE

Figure 8.2

It seemed that she developed a sense of how she could make use of the numbers she had already used, in completing her project, while continuing to work in an exploratory mode.

By the time she completed this project, Harriet had mastered the use of 90, 30 and 180 degrees, had learned how to use symmetry, had mastered the basic idea of procedures, subprocedures and superprocedures, and had a good understanding of how to use error messages for debugging.

Harriet's next project in Turtle Geometry arose out of some explorations she was doing with SPIN commands. Combining SPINs with forwards and circles, she got the idea of making a cowboy, twirling a rope. I suggested that this would make a nice, although difficult project. Harriet made the cowboy quite nicely, and then decided to put a hat on his head. Since the TURTLE was spinning at the end of a loop of rope, she had to be shown how to reverse the steps of SPINs, turns and forwards, to get the TURTLE back to where it started, so that she could add the hat. Harriet needed help with debugging TURTLE state problems several times during this project. She was introduced to the idea of "playing computer," going through a procedure step by step, and encouraged to number her steps by fives, and to use subprocedures. She chose not to use subprocedures for the COWBOY, but did use subprocedures to draw and locate the HAT. (Figure 8.3)

```
TO LASSO
1 COWBOY
2 BLACK
3 HAT
END
```

```
TO COWBOY
1 PENUP
2 FORWARD 100
3 PENDOWN
4 RCIRCLE 30
5 LEFT 180
6 LARC 30
7 RIGHT 90
8 FORWARD 200
9 PENUP
10 RIGHT 180
11 FORWARD 100
12 RIGHT 90
13 PENDOWN
14 FORWARD 50
15 BACK 50
16 LEFT 180
17 FORWARD 50
18 SPIN 100
19 FORWARD 30
20 SPIN 500
21 FORWARD 30
22 RCIRCLE 30
END
```

```
TO BLACK
1 BACK 30
2 SPIN - 500
3 BACK 30
4 SPIN - 100
5 BACK 50
6 RIGHT 90
END

TO HALF
1 RARC 15
2 RARC 15
END
```

```
TO HAT
1 STRATE
2 HALF
END

TO STRATE
1 PENUP
2 FORWARD 160
3 RIGHT 90
4 PENDOWN
5 FORWARD 30
6 LEFT 180
7 FORWARD 60
8 RIGHT 180
9 FORWARD 20
10 LEFT 90
END
```

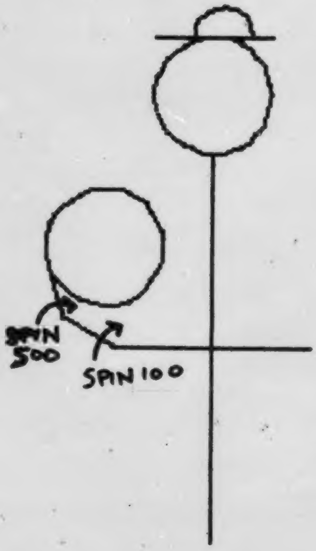
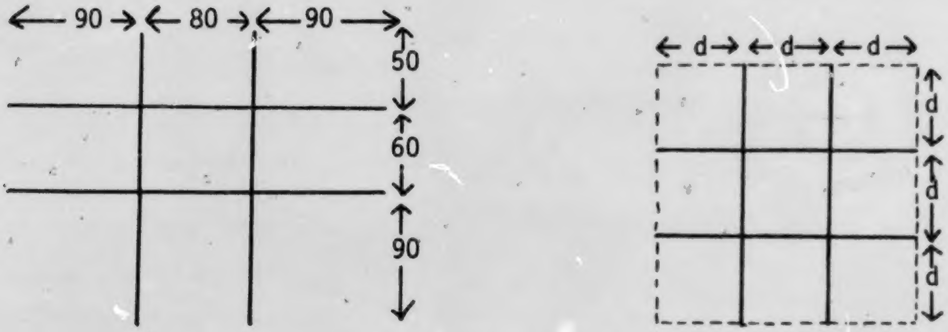


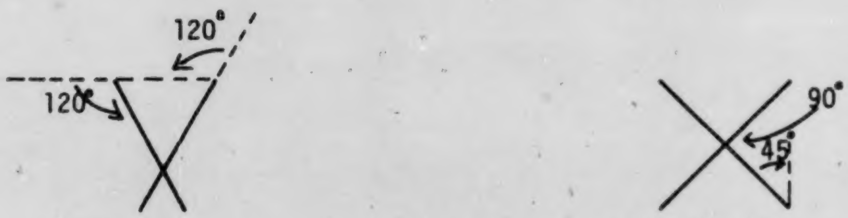
Figure 8.3

The rest of Harriet's Turtle Geometry work consisted of small projects: some spinning designs, an oval, a clock, and a TICTACTOE board that turned into her next major project. In working on her TICTACTOE board, Harriet made use of the same kind of exploratory approach -- combined with a lack of overall analysis -- that she used in her SMILY project. Her final drawing of a TICTACTOE board, made use of horizontal, but not vertical symmetry, and had uneven spacing. Harriet's idea of what a TICTACTOE board looks like seems clear from analysis of her dribble file: two lines crossing two other lines at right angles. She did not have the idea of the board consisting of "nine squares", which would have led to a symmetrical drawing, and simplified the trial and error process of making the procedure. Similarly, in making her "X", for use in the TICTACTOE game, Harriet worked by a process of trial and error, to get a somewhat asymmetrical X, rather than use an analytical approach which could have produced a symmetrical X.



Harriet's TicTacToe Board

"Idealized" TicTacToe Board



Harriet's X

"Idealized" X

Figure 8.4

In making her X, Harriet did make use of 30, 60 and 90 degree angles, eventually combining them into 120 degree angles, which she used at both top corners of the X.

8.3 Harriet's Work on Major Interactive Projects

8.3.1 Harriet's TICTACTOE Project

The TICTACTOE project lasted for seven class sessions. Harriet had expressed a sense of boredom with Turtle Geometry, and a curiosity about what else the computer could do. When she used the TURTLE to draw a tictactoe board, the teacher suggested that she develop this idea further, making a game which could be played by two people, in which the computer would draw the game, and keep track of the score. A possibility of making the computer one of the players was put off until later.

In order to carry out this project, Harriet had to learn a number of new ideas about LOGO:

the use of TYPE and PRINT commands
the use of MAKE to name points and variables
the use of REQUEST to get inputs from the players
the use of SETT to place the TURTLE at the right spot to
    draw an X or O

the use of IF and =, to determine where the next move should go; as
    well as the status of the game
the use of lists to keep track of moves and of winning combinations
the use of SENTENCE to add new elements to a list
setting up a data structure to keep track of the game
functional names for subprocedures

Since all this new material was difficult for Harriet to absorb, the teacher introduced new ideas to her in smallish doses, as she expressed a need for them. First the procedure would be talked through, to elicit as much description from her as possible, of what she wanted the computer to do. After talking an idea through verbally, she would be shown how to translate it into LOGO, establishing the pattern that should be followed, writing down the steps and sequences in her notebook. Harriet then would copy them with careful attention to detail.

Harriet functioned extremely well in this mode. Although she did not fully understand what she was doing, she felt confident that she could get the bugs worked out. Harriet was an excellent typist, and made few typing errors. When she did need to correct, revise or add to a line, she easily learned to use EDL and CTRL-N, as editing aids.

Problems began to occur when Harriet started to make the program intelligent enough to keep track of moves and record a win by either player, or a tie. Here she ran into difficulty because she was a bit confused about the overall structure of her programs -- not always sure where a particular subprocedure should be put -- into PLAY1 and PLAY2, her key subprocedures, or into a sub-sub-procedure. In such a case, Harriet usually needed help to locate the source of a bug. Once a bug was found by "playing computer" Harriet seemed to understand what had caused it and how it could be fixed.

Considerations of one bug in detail will illustrate this process. Near the end of the project, Harriet had completed her game to the point where it would announce a winner, draw a line through the three Xs or Os, and stop. If the game was a tie, however, the computer would just keep playing, asking for additional moves. Harriet was asked how she thought the computer could keep track of whether the game was a tie. She suggested having the computer count the number of moves. If the total number of moves was nine, the computer would declare the game a tie, and stop.

In implementing this idea, Harriet was reminded how to initialize the variable "COUNT, by using MAKE "COUNT 0, and how to increase it using MAKE "COUNT :COUNT+1. The problem was, where to put these instructions, and the final game ending instruction, IF :COUNT = 9 TOPLEVEL. In talking it through, she realized that the initialization of "COUNT should go into the superprocedure, TICTACTOE, and that the increases should go into the PLAY1 and PLAY2 procedures. Harriet also realized that the ending condition only needed to be put into PLAY1, because the value of "COUNT could only be 9 during the first player's turn.

What Harriet forgot, was to put the line MAKE "COUNT :COUNT+1, into PLAY2 as well as PLAY1. When the count condition did not work, Harriet could not debug this by herself. When it was suggested that she have the computer PRINT the value of COUNT, the result was 5, indicating that only the odd moves had been counted. Then printing out PLAY2, Harriet realized that MAKE "COUNT :COUNT+1 was missing and she added it at the appropriate place with no difficulty.

```
TO TICTACTOE
5 MAKE "COUNT 0
10 TAC
12 MAKE "LS [ ]
13 MAKE "LI [ ]
20 PRINT [WHO IS PLAYER ONE?]
30 MAKE "P1 REQUEST
40 PRINT [WHO IS PLAYER TWO?]
50 MAKE "P2 REQUEST
60 PLAY1
70 PLAY2
80 GO 60
85
END
```

```
TO PLAY1
5 MAKE "COUNT :COUNT +1
20 TYPE :P1
30 PRINT ['S TURN TO MOVE]
40 PRINT [WHICH BOX WOULD YOU LIKE TO PUT AN X IN?]
50 MAKE "CHOICE REQUEST
55 MAKE "LS ( SENTENCE :LS :CHOICE )
60 MAKEX
65 CHECKWINNER1
70 IF :COUNT = 9 PRINT [IT IS A TIE] TOPLEVEL
END

TO PLAY2
5 MAKE "COUNT :COUNT +1
10 TYPE [IT IS]
20 TYPE :P2
30 PRINT ['S TURN TO MOVE]
40 PRINT [WHICH BOX WOULD YOU LIKE TO PUT AN O IN?]
50 MAKE "LI ( SENTENCE :LI :CHOICE )
60 MAKEO
65 CHECKWINNER2
70 IF :COUNT = 9 PRINT [IT IS A TIE] TOPLEVEL
END
```

```
TO TAC
5 PENUP
10 BACK 60
15 PENDOWN
20 FORWARD 200
25 PENUP
30 RIGHT 90
35 FORWARD 80
40 RIGHT 90
45 PENDOWN
50 FORWARD 200
55 PENUP
60 RIGHT 90
65 FORWARD 200
70 RIGHT 90
75 FORWARD 150
80 RIGHT 90
85 FORWARD 30
90 PENDOWN
95 FORWARD 260
100 RIGHT 90
105 PENUP
110 FORWARD 60
115 RIGHT 90
120 PENDOWN
125 FORWARD 260
END
```

```
TO 0 :POINT
10 PENUP
20 SETT :POINT
30 PENDOWN
40 RCIRCLE 20
END
```

```
TO X :POINT
10 PENUP
20 SETT :POINT
30 PENDOWN
40 EX
END
```

```
TO MAKEX
10 IF :CHOICE = [1] X :H1
20 IF :CHOICE = [2] X :H2
30 IF :CHOICE = [3] X :H3
40 IF :CHOICE = [4] X :H4
50 IF :CHOICE = [5] X :H5
60 IF :CHOICE = [6] X :H6
70 IF :CHOICE = [7] X :H7
80 IF :CHOICE = [8] X :H8
90 IF :CHOICE = [9] X :H9
END
```

```
TO MAKEO
10 IF :CHOICE = [1] 0 :H1
20 IF :CHOICE = [2] 0 :H2
30 IF :CHOICE = [3] 0 :H3
40 IF :CHOICE = [4] 0 :H4
50 IF :CHOICE = [5] 0 :H5
60 IF :CHOICE = [6] 0 :H6
70 IF :CHOICE = [7] 0 :H7
80 IF :CHOICE = [8] 0 :H8
90 IF :CHOICE = [9] 0 :H9
END
```

```
TO EX
5 RIGHT 30
10 FORWARD 30
15 LEFT 120
20 PENUP
25 FORWARD 20
30 LEFT 120
35 PENDOWN
40 FORWARD 30
END
```

```
TO CHECKWINNER1
10 IF ALLOF :Z1 :LS LINE1 PW1
20 IF ALLOF :Z2 :LS LINE2 PW1
30 IF ALLOF :Z3 :LS LINE3 PW1
40 IF ALLOF :Z4 :LS LINE4 PW1
50 IF ALLOF :Z5 :LS LINE5 PW1
60 IF ALLOF :Z6 :LS LINE6 PW1
70 IF ALLOF :Z7 :LS LINE7 PW1
80 IF ALLOF :Z8 :LS LINE8 PW1
END

TO CHECKWINNER2
10 IF ALLOF :Z1 :LI LINE1 PW2
20 IF ALLOF :Z2 :LI LINE2 PW2
30 IF ALLOF :Z3 :LI LINE3 PW2
40 IF ALLOF :Z4 :LI LINE4 PW2
50 IF ALLOF :Z5 :LI LINE5 PW2
60 IF ALLOF :Z6 :LI LINE6 PW2
70 IF ALLOF :Z7 :LI LINE7 PW2
80 IF ALLOF :Z8 :LI LINE8 PW2
END
```

```
TO LINE1
10 PENUP
20 SETT :H1
30 PENDOWN
40 FORWARD 200
END

TO LINE2
10 PENUP
20 SETT :H6
30 RIGHT 180
40 RIGHT 180
50 FORWARD 20
END
```

```
TO LINE5
10 PENUP
20 SETT :H1
22
30 RIGHT 28
40 PENDOWN
50 FORWARD 230
END

TO LINE6
10 PENUP
20 SETT :H3
30 RIGHT 90
40 PENDOWN
50 FORWARD 170
END
```

```
TO LINE3
10 PENUP
20 SETT :H7             TO LINE7
30 PENDOWN              10 PENUP
40 FORWARD 200          20 SETT :H2
END                     30 RIGHT 90
                        50 PENDOWN
TO LINE4                60 FORWARD 170
10 PENUP                END
20 SETT :H3
30 PENDOWN              TO LINE8
40 RIGHT 140            10 PENUP
50 FORWARD 200          20 SETT :H1
END                     30 RIGHT 90
                        40 PENDOWN
                        50 FORWARD 170
                        END

TO PW1
10 PRINT ( SENTENCE :P1 [IS THE WINNER])
20 TOPLEVEL
END

TO PW2
10 PRINT ( SENTENCE :P2 [IS THE WINNER])
20 TOPLEVEL
65 CHECKWINNER2
END

TO ALLOF :LIST1 :LIST2
10 IF BOTH PARTOF FIRST :LIST1 :LIST2 BOTH PARTOF FIRST BUTFIRST :LIST1
  :LIST2 PARTOF LAST :LIST1 :LIST2 OUTPUT "TRUE
20 OUTPUT "FALSE
END

TO PARTOF :ITEM :LIST2
10 IF :LIST2 = [] OUTPUT "FALSE
20 IF FIRST :ITEM = FIRST :LIST2 OUTPUT "TRUE
30 OUTPUT PARTOF :ITEM BUTFIRST :LIST2
END
```

8.3.2 Harriet's MADLIB Project

After completing her TICTACTOE project, Harriet was at a loss for what to do next. She was offered a choice of six different projects. The idea of doing a "madlib" story was the most appealing to her. The basic idea is simple. The computer prints out a simple story, substituting randomly chosen nouns, verbs, adjectives, etc., for specific words in the story. This project offered many fewer "new learnings", for Harriet, and so was one that she could help her consolidate what she had already learned. The new ideas she had to learn were these:

--procedures with outputs

--the use of RANDOM to choose a number which would determine the particular word to be OUTPUT from a preset list

--the use of % to create spaces in TYPED output

--the distinction between TYPE and PRINT

Once Harriet was shown the syntax for using RANDOM and OUTPUT, she was able to carry through the project almost entirely on her own. The structure of the project was much simpler than that of the TICTACTOE project, so that when bugs occured, Harriet could usually fix them herself. She had a little difficulty understanding that she had to provide an output for each number, 0 through 9, output by RANDOM. In some of her procedures she listed 10 rather than 0, and would sometimes get a confusing "NOUN DIDN'T OUTPUT AT LEVEL 1 LINE -- in MADLIB." This message was confusing to Harriet because it did not occur consistently (approximately once out of every ten tries). NOUN appeared to work perfectly some of the time and not work at all at other times. As each of her subprocedures was debugged, Harriet gradually came to understand the cause of this occasional problem.

Another more straightforward bug in the MADLIB procedure involved the length of each statement involving a PRINT or TYPE command. Since the randomly chosen words had varying lengths, the typed lines would sometimes run over the edge of the page, causing a word to be split between two lines. Harriet began the process of eliminating these over-runs, adjusting the length of each line by changing the location of PRINT and TYPE commands. Of course, shortening the first line made the second longer, so that the process had to be repeated for each succeeding line. Harriet had not completed this when the series of classes ended.

```
?MADLIB
ONCE UPON A HULA HOOP, THERE LIVED A CREAMY GIRL NAMED
DAN.  SHE HAD A WICKED STEPMOTHER AND 2 SMELLY STEPSISTERS WHOSE
    NAMES WERE EGBERT AND GRISELDA.  CINDERELLA HAD TO  DO ALL THE
TWISTED WORK WHILE THE 2 CHEESE RAVIOLI'S WERE LAZY.  ONE DAY ALL THE

PEOPLE OF THE KINGDOM WERE INVITED TO PRINCE GERTRUDE'S SMELLY BAL
L.  CINDERELLA  COULDN'T GO.  BUT HER 2 STEPSISTERS & STEP
CHEESE RAVIOLI
COULD.  EGBERT THE FAIRY GODMOTHER CAME & GAVE CINDERELLA A ICE AND A
TOILET BOWL SO SHE COULD GO TO THE PINE CONE.  THE
CHEESE RAVIOLI FELL IN LOVE WITH HER.  SHE TICK, OUT OF THE CIGAR
   AND LOST HER  CREAMY , TOILET BOWL.  THE PRINCE LOOKED FOR HER TO PU
T THE
GRANDMA BACK ON.  HE FOUND HER AND THEY LIVED
SILENTLY EVER AFTER.
```

```
TO MADLIB
10 TYPE [ONCE UPON A%]
20 TYPE NOUN
30 TYPE [, THERE LIVED A%]
40 TYPE ADJETIVE
50 PRINT [%GIRL NAMED]
60 TYPE [. SHE HAD A WICKED STEPMOTHER AND 2%]
70 TYPE ADJETIVE
80 PRINT [% STEPSISTERS WHOSE %]
85 PRINT [% NAMES WERE%]
90 TYPE N
100 TYPE [% AND GRISELDA. CINDERELLA HAD TO %]
105 PRINT [% DO ALL THE %]
110 TYPE ADJETIVE
120 TYPE [% WORK WHILE THE 2%]
130 TYPE NOUN TYPE  2%]
140 PRINT [% WERE LAZY. ONE DAY ALL THE %] TYPE [%PEOPLE OF THE%]
145 TYPE [% KINGDOM WERE INVITED TO PRINCE%]
150 TYPE N TYPE ['S %] TYPE ADJETIVE
160 TYPE [% BALL. CINDERELLA %]
165 TYPE [% COULDN'T GO, BUT HER 2 STEPSISTERS & STEP%]
170 PRINT NOUN
180 TYPE [COULD %}
190 TYPE N
200 TYPE [% THE FAIRY GODMOTHER CAME & GAVE CINDERELLA A%]
210 TYPE NOUN
220 PRINT [%AND A%]
230 TYPE NOUN
240 TYPE [%SO SHE COULD GO TO THE%]
250 TYPE NOUN
260 PRINT [ %THE%]
270 TYPE NOUN
280 TYPE [%FELL IN LOVE WITH HER.  SHE%]
290 TYPE VERB TYPE [,]
300 TYPE [% OUT OF THE %]
310 PRINT NOUN
320 TYPE [% AND LOST HER %]
330 TYPE ADJETIVE TYPE [% , %] TYPE NOUN
340 PRINT [ % THE PRINCE LOOKED FOR HER TO PUT THE %]
350 TYPE NOUN
360 PRINT [% BACK ON.  HE FOUND HER AND THEY LIVED %]
370 TYPE ADVERB
```

```
380 PRINT [% EVER AFTER %]
END

TO N
10 MAKE "NUMBER RANDOM
20 IF :NUMBER = 1 OUTPUT [CONGA]
30 IF :NUMBER = 2 OUTPUT [EGBERT]
40 IF :NUMBER = 3 OUTPUT [ISMEENEE]
50 IF :NUMBER = 4 OUTPUT [GERTRUDE]
60 IF :NUMBER = 5 OUTPUT [GERSHOM]
70 IF :NUMBER = 6 OUTPUT [DAN]
80 IF :NUMBER = 7 OUTPUT [TWEEDLEDUM]
90 IF :NUMBER = 8 OUTPUT [BALLYHOO]
100 IF :NUMBER = 9 OUTPUT [DAVID]
110 IF :NUMBER = 0 OUTPUT [WILLIMENA]

TO NOUN
10 MAKE "NUMBER RANDOM
20 IF :NUMBER = 1 OUTPUT [TRASH CAN]
30 IF :NUMBER = 2 OUTPUT [CIGAR]
40 IF :NUMBER = 3 OUTPUT [HULA HOOP]
50 IF :NUMBER = 4 OUTPUT [GRANDMA]
60 IF :NUMBER = 5 OUTPUT [ICE]
70 IF :NUMBER = 6 OUTPUT [TOOTHPASTE]
80 IF :NUMBER = 7 OUTPUT [TOILET BOWL]
90 IF :NUMBER = 8 OUTPUT [MUD]
100 IF :NUMBER = 9 OUTPUT [PINE CONE]
110 IF :NUMBER = 0 OUTPUT [CHEESE RAVIOLI]
END
```

```
TO ADJETIVE
10 MAKE "NUMBER RANDOM
20 IF :NUMBER = 1 OUTPUT [UGLY]
30 IF :NUMBER = 2 OUTPUT [PURPLE]
40 IF :NUMBER = 3 OUTPUT [SMELLY]
50 IF :NUMBER = 4 OUTPUT [FLUFFY]
60 IF :NUMBER = 5 OUTPUT [CREAMY]
70 IF :NUMBER = 6 OUTPUT [TWISTED]
80 IF :NUMBER = 7 OUTPUT [HARD]
90 IF :NUMBER = 8 OUTPUT [WRINKLED]
100 IF :NUMBER = 9 OUTPUT [BILLOWY]
110 IF :NUMBER = 0 OUTPUT [ITCHY]
END

TO ADVERB
10 MAKE "NUMBER RANDOM
17 IF :NUMBER > 5 OUTPUT [SILLY]
20 IF :NUMBER = 1 OUTPUT [CALMY]
30 IF :NUMBER = 2 OUTPUT [STUPIDLY]
40 IF :NUMBER = 3 OUTPUT [ANNOYINGLY]
50 IF :NUMBER = 4 OUTPUT [PATHETICLY]
60 IF :NUMBER = 5 OUTPUT [SILENTLY]
70 IF :NUMBER = 0 OUTPUT [GENTLY]
END

TO VERB
10 MAKE "NUMBER RANDOM
20 IF :NUMBER = 1 OUTPUT [WALK]
30 IF :NUMBER = 2 OUTPUT [YELL]
40 IF :NUMBER = 3 OUTPUT [SING]
50 IF :NUMBER = 4 OUTPUT [SKIP]
60 IF :NUMBER = 5 OUTPUT [ROCK]
70 IF :NUMBER = 6 OUTPUT [ROLL]
80 IF :NUMBER = 7 OUTPUT [TICK]
90 IF :NUMBER = 8 OUTPUT [GIGGLE]
100 IF :NUMBER = 9 OUTPUT [SHIT]
110 IF :NUMBER = 0 OUTPUT [BLEED]
END
```

8.4 Conclusion

Harriet was one of the most competent of the children in our sample groups at using the computer keyboard, understanding the computer's possibilities, and at using diverse elements of the LOGO language appropriately to achieve her purposes. At the same time, her sense of "creative involvement" was minimal. The more creative her project, the more tedius typing and debugging were required. She did not enjoy the process of finding and eliminating bugs, the way that Gary, a student in one of the earlier classes did.

After her last class, Harriet was asked whether she would like to work with a computer again. "Well," she reflected, "it would depend on what the other choices were."

9. Jimmy

Jimmy is considered to be an above average student by his teachers. His national achievement test scores place him in the 57th percentile. Although he is considered bright, Jimmy is extremely selective in applying himself to regular class work. He tends to excel in areas which appeal to him, and offers resistance to assignments which he does not consider interesting.

1. Jimmy's Working Style

In the LOGO classes, his work was also somewhat ideosyncratic. He quickly became expert at "TURTLE driving," and created one of the most elaborate drawings of any of our students. His robot procedure, patterned on the Star Wars character, R2D2, was extremely detailed and intricate, and was drawn with uncommon care and patience. On the other hand, Jimmy had great difficulty understanding how to use subprocedures to make his work easier. He conceived of each project as a long, almost endless string of commands; subprocedures were just a way of breaking a project into pieces. It was not until near the end of the classes that Jimmy began to make use of subprocedures for planning. Debugging remained a mystery for him.

2. Jimmy's Sailboat Project

Jimmy's first project, after learning the basic LOGO commands, was to use the TURTLE to draw a sailboat. He drew his boat step by step on the screen, and copied the entire list of commands to make the procedures, BODY and SAIL. Although he copied steps literally, in a style similar to Deborah's, he was also able to make good use of geometric knowledge. Lines 2, 3, 4, and 5 of his BODY procedure, which draw the prow of his boat, show how he used a 180 degree angle to reverse the TURTLE's direction (Figure 9.1a). Line's 12 and 13 show how he corrected a mistaken RIGHT 89, by doubling 89, and following it by the command LEFT 178. It is characteristic of Jimmy's work that he did not replace both commands by one correct command, LEFT 89, but kept both his original commands as part of his procedure.

```
TO BOAT
1 BODY
2 FORWARD 20
3 SAIL
4 HIDETURTLE
END
```

```
TO BODY
1 LEFT 90
2 FORWARD 60
3 RIGHT 40
4 FORWARD 20
5 RIGHT 140
6 FORWARD 70
7 FORWARD 10
8 RIGHT 96
9 FORWARD 14
10 RIGHT 180
11 FORWARD 14
12 RIGHT 89
13 LEFT 178
14 FORWARD 20
15 LEFT 10
END
```

```
TO SAIL
1 RIGHT 90
2 FORWARD 60
3 RIGHT 140
4 FORWARD 50
5 FORWARD 20
6 RIGHT 100
7 RIGHT 20
8 RIGHT 10
9 FORWARD 50
10 RIGHT 10
11 LEFT 10
12 FORWARD 30
13 RIGHT 100
14 RIGHT 20
15 LEFT 10
16 RIGHT 5
17 FORWARD 40
18 FORWARD 20
19 RIGHT 80
20 FORWARD 10
END
```

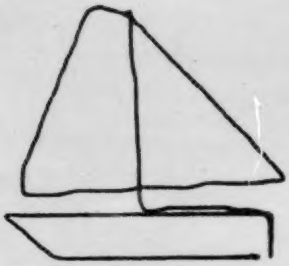


Figure 9.1

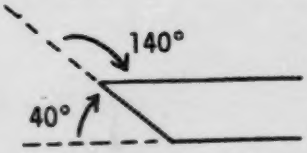


Figure 9.1a

3. Jimmy's next major project, which he worked on for five class periods (approximately 6 1/2 hours), was his drawing of a robot (figure 9.2).

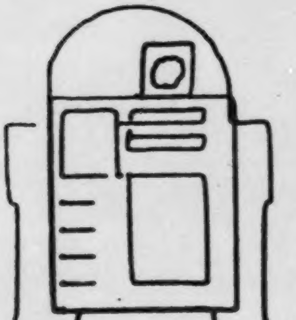
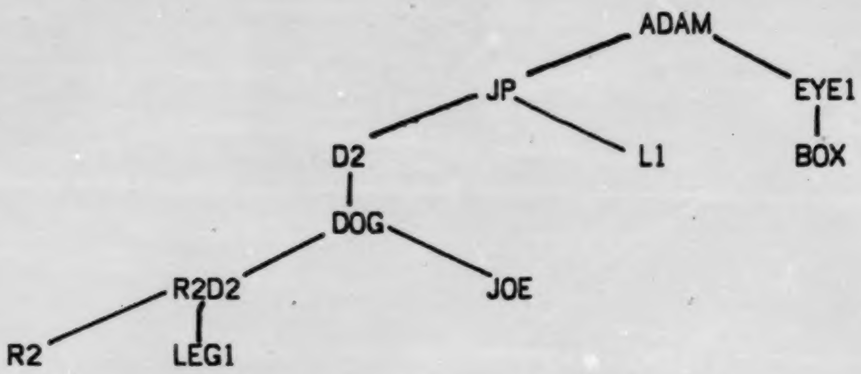


Figure 9.2

Jimmy's superprocedure, ADAM, made use of ten different subprocedures, to draw the robot. Jimmy worked on the project in an exploratory fashion. When he had successfully drawn one part of the robot on the screen, he would write down all the steps on the text display screen, and give that part of his drawing a name. When he began working again, he would include the procedure he had just defined, as the first step of his next procedure. Jimmy's procedure hierarchy can be described as a "tree", with no branch containing more than two different sub-branches.



The role of each of the subprocedures can be traced by starting at the bottom of the tree, and working upward (see Figure 9.2):

R2 draws the rectangle which made up most of the robot's body.

LEG1 draws the right leg.

R2D2 includes R2 and LEG1 and then adds the left leg.

JOE draws one 30-by-5 rectangle on the right of the robot.

DOG includes R2D2 and JOE and adds another small rectangle.

D2 includes DOG and adds the two larger rectangles.

L1 draws a single line on the lower left of the robot

JP includes D2 and four repeats of L1. It draws the entire robot, except for the eye.

BOX draws a small square in the robot's head.

EYE1 includes BOX and draws the eye in the center of the box.

ADAM includes JP and EYE1. It draws the entire robot, moves the TURTLE over and draws the eye.

The specific subprocedures are listed below:

```
TO R2
1 FORWARD 80
2 RIGHT 90
3 FORWARD 70
4 RIGHT 90
5 FORWARD 80
6 RIGHT 90
7 FORWARD 70
8 RIGHT 90
9 FORWARD 80
10 RARC 40
11 RARC 40
END
```

```
TO LEG1
1 FORWARD 10
2 LEFT 90
3 FORWARD 15
4 RIGHT 90
5 FORWARD 30
6 RIGHT 90
7 FORWARD 5
8 LEFT 90
9 FORWARD 40
10 LEFT 40
11 RIGHT 10
12 RIGHT 5
13 FORWARD 30
14 RIGHT 100
15 RIGHT 20
16 LEFT 10
17 FORWARD 20
18 RIGHT 90
19 FORWARD 20
20 RIGHT 180
21 FORWARD 20
22 RIGHT 90
23 FORWARD 10
24 RIGHT 90
25 FORWARD 30
END
```

```
TO R2D2
5 R2
10 LEG1
15 LEFT 90
20 RIGHT 15
22 LEFT 10
25 FORWARD 50
30 LEFT 90
35 FORWARD 30
40 RIGHT 90
45 FORWARD 10
50 RIGHT 90
55 FORWARD 20
60 RIGHT 180
65 FORWARD 20
70 RIGHT 90
75 FORWARD 15
80 RIGHT 100
85 FORWARD 30
90 LEFT 10
95 FORWARD 40
100 RIGHT 90
105 LEFT 180
110 FORWARD 5
115 RIGHT 90
120 FORWARD 30
125 RIGHT 90
130 FORWARD 15
END
```

```
TO JOE
1 PENUP
2 FORWARD 30
3 PENDOWN
4 FORWARD 30
5 LEFT 90
6 FORWARD 5
7 LEFT 90
8 FORWARD 30
9 LEFT 90
10 FORWARD 5
END
```

```
TO DOG
1 R2D2
2 JOE
3 PENUP
4 FORWARD 5
5 PENDOWN
6 FORWARD 5
7 LEFT 90
8 FORWARD 30
9 LEFT 90
10 FORWARD 5
11 LEFT 90
12 FORWARD 30
END
```

```
TO L1
5 FORWARD 15
10 BACK 15
15 PENUP
20 RIGHT
25 FORWARD 10
30 PENDOWN
35 LEFT 90
END
```

```
TO D2
1 DOG
2 RIGHT 90
3 RIGHT 180
4 PENUP
5 FORWARD 15
6 LEFT 90
7 PENDOWN
8 FORWARD 30
9 RIGHT 90
10 FORWARD 40
11 RIGHT 90
12 FORWARD 30
13 RIGHT 90
14 FORWARD 40
15 LEFT 90
16 PENUP
17 FORWARD 5
18 RIGHT 90
19 PENDOWN
20 FORWARD 25
21 LEFT 90
22 FORWARD 20
23 LEFT 90
24 FORWARD 25
25 LEFT 90
26 FORWARD 20
END
```

```
TO JP
1 D2
2 BACK 20
3 RIGHT 90
4 PENUP
5 FORWARD 10
6 LEFT 90
7 PENDOWN
8 L1
9 L1
10 L1
11 L1
12 HIDETURTLE
END
```

```
TO EYE1
5 BOX
10 PENUP
15 LEFT 90
20 FORWARD 10
25 LEFT 90
30 FORWARD 4
35 RIGHT 90
40 PENDOWN
45 LCIRCLE 10
END
```

```
TO BOX
5 FORWARD 20
10 LEFT 90
15 FORWARD 20
20 LEFT 90
25 FORWARD 20
30 LEFT 90
35 FORWARD 20
END
```

```
TO ADAM
5 JP
10 SHOWTURTLE
15 PENUP
20 LEFT 90
25 FORWARD 80
30 RIGHT 90
35 FORWARD 30
40 PENDOWN
45 EYE1
50 HIDETURTLE
END
```

4. Jimmy's Rocket Project

Jimmy's next major project, although visually much simpler than the robot, also took five class periods to work out. The complexity of this project lay in the use of angles other than ninety degrees. Jimmy had to pay careful attention to TURTLE's state at each point of the process, in order to draw a symmetrical design. (Figure 9.3)

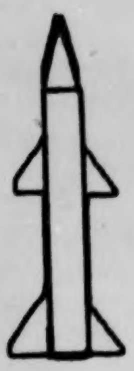
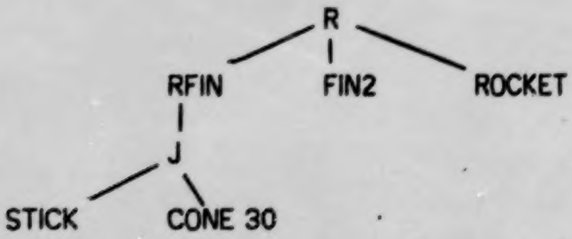


Figure 9.3

Jimmy's hierarchy "tree" was a lot simpler this time:



STICK draws the rocket's body.

CONE 30 draws the nose cone.

J combines CONE and STICK and draws the entire rocket.

RFIN includes J and adds the upper right fin.

FIN2 draws the lower right fin.

ROCKET draws both left fins.

R includes RFIN, FIN2 and ROCKET, and draws the entire rocket.

In drawing the rocket, Jimmy constructed the upper right fin by trial and error.

He then copied it three more times to draw the three other fins, paying careful attention to right/left symmetry. (The lower left fin came out a little too large because of a miscalculation).

Jimmy developed an interesting solution to the problem of drawing the rocket's nose cone -- an isoceles triangle with a fixed base. His strategy was to start at point A (figure 9.4), turn the TURTLE RIGHT 15 degrees, move it forward a certain distance, and then back the same distance. Making use of symmetry, Jimmy then moved the TURTLE to point B, turned the TURTLE <u>LEFT</u> 15, and moved it forward and back the <u>same</u> distance. His problem was that he did not know the distance to use. The teacher suggested that he make that length a variable, and together they talked through the procedure:

```
TO CONE :LEN
5 FORWARD 100
10 RIGHT 15
15 FORWARD :LEN
20 BACK :LEN
25 RIGHT 75
30 FORWARD 15
35 LEFT 90
40 LEFT 15
45 FORWARD :LEN
50 BACK :LEN
55 RIGHT 15
END
```

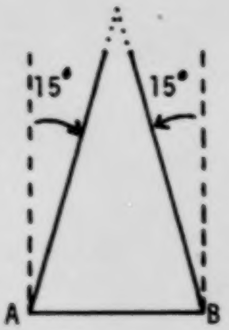


Figure 9.4

Jimmy then used the CONE procedure with several different inputs, until he found a length that looked right to him. He used this length CONE 30 in his procedure, J, which drew the basic rocket body.

There are clues in the CONE procedure to the way that Jimmy made use of "units" of 90 degrees in thinking about his drawing. He turned the TURTLE RIGHT 15 at line 10. In line 25 he added the additional rotation, RIGHT 75, to make a total of 90 degrees, allowing the TURTLE to move from point A to point B. Having reached point B, Jimmy turned the TURTLE LEFT 90, and added the step, LEFT 15. The fact that he was thinking in terms of 90^0 units was confirmed by his failure to combine the last two as one step, LEFT 105.

5. Jimmy's Racetrack Project

During the last few classes, Jimmy concentrated on Dynaturtle activities. Part of his work with Dynaturtle is described in detail in Chapter 6 of Part II of this report. A racetrack project that Jimmy carried out in connection with dynamic activities gives another indication of the way in which he incorporated geometric ideas into his work in Turtle Geometry.

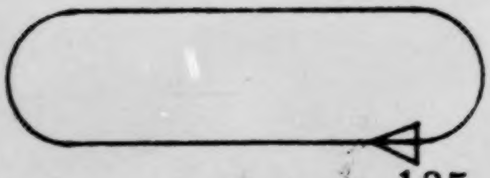
Jimmy was drawing a "racetrack" for the Dynaturtle. He decided to draw an oval track, and had completed drawing the inside of the track. To draw the ends of the inside of the track, Jimmy made repeated use of RARC 40 commands (Figure 9.5). To draw the outside of the track, he moved the TURTLE over 40 TURTLE steps, and drew the outside using RARC 80 commands (Figure 9.6). When an observer asked Jimmy how he knew that RARC 80 would draw the correct sized arcs, Jimmy answered that using an RARC 80 command was "just like making a corner with sides of 80." This indicated that he understood how to replace the effect of the RARC command by an equivalent one which made it easy for him to calculate the necessary distance (Figure 9.7).

```
TO TRACK
10 IN
20 OUT
END
```

```
TO IN
10 RIGHT 90
20 FORWARD 50
30 LEFT 180
40 FORWARD 150
50 RARC 40
60 RARC 40
70 FORWARD 150
80 RARC 40
90 RARC 40
100 PENUP
110 LEFT 90
120 FORWARD 40
130 PENDOWN
END
```

```
TO OUT
10 RIGHT 90
20 FORWARD 150
30 RARC 80
40 RARC 80
50 FORWARD 150
60 RARC 80
70 RARC 80
END
```

Figure 9.5



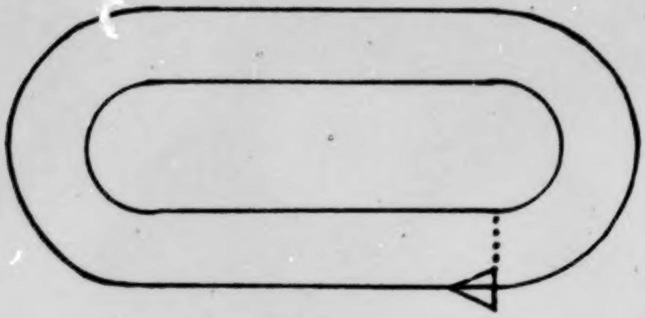


Figure 9.6

Figure 9.7a

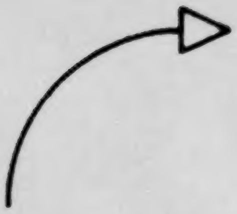
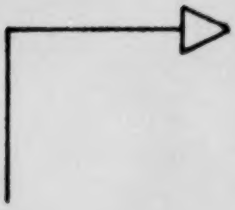


Figure 9.7b



10. Karl

Karl is a boy who has severe learning difficulties in standard school subjects. He is several years behind his peers in basic academic skills, and has been diagnosed as having a "learning disability". He receives regular individual assistance in the school's Learning Center. Karl is large for his age and somewhat awkward in manner. He has few friends among his classmates, other than a few selected "cronies".

When Karl was interviewed, prior to his participation in LOGO some significant gaps occurred between his abilities in tasks that required reading, writing, arithmetic, and verbal expression, and those that required non-verbal problem solving. While appearing to be virtually unable to read or carry out arithmetic calculations, he readily solved problems involving number patterns and attributes of different shapes, and was able to systematically generate all twenty-four permutations of a set of cubes of four different colors. Questions that were designed to elicit Karl's feelings about school, and about himself and friends, were answered with statements like: "I don't care", "Nothing", "No", and "I don't know".

Despite extreme difficulty using the computer keyboard, Karl did very well in the LOGO classes. From the beginning he enjoyed the sense of control he felt while participating in LOGO activities. While Karl often needed help remembering the spelling of various commands, he was able to make use of his logical and organizational abilities in a variety of contexts. Karl's accomplishments in the LOGO classes regularly surpassed what he was normally considered able to do in his regular classes and in his special tutoring sessions. At present, Karl's teachers are trying to decide how to take into account the information learned during his successful LOGO experience, in planning his regular school work.

In the sections that follow, we will review Karl's major accomplishments, as well as some of the difficulties that he experienced.

10.1 Karl's Major Accomplishments

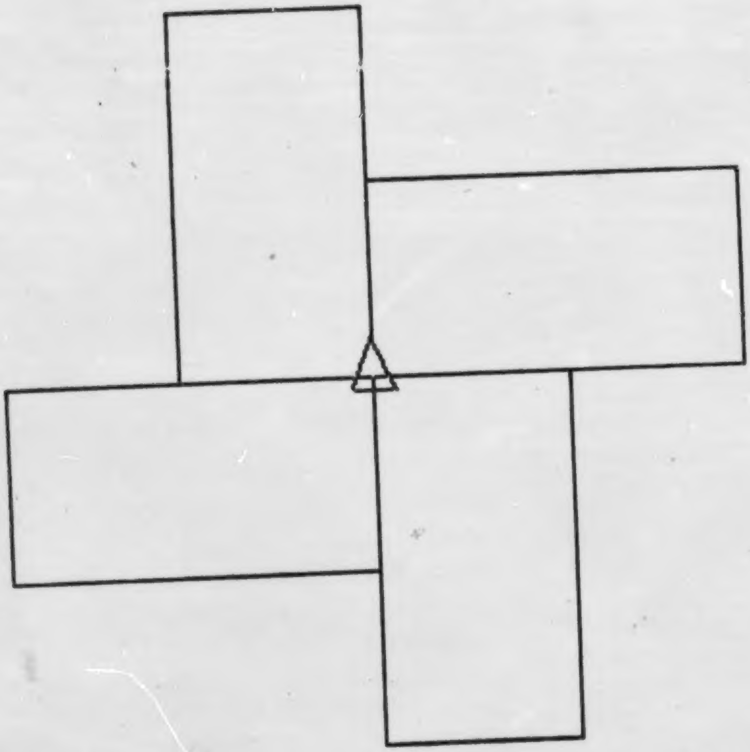
Karl's accomplishments in the LOGO classes fit into two categories: he developed a processes both for working with the computer and for planning and debugging his work, and he completed a number of successful projects in the area of Turtle Geometry, culminating in a keyboard controlled animation, which allowed him to drive the TURTLE around the screen.

At first Karl's work alternated between carefully planned designs and the haphazard use of commands that he did not really understand. As he began to discover consistencies in the effects produced by different commands, he gradually came to exercise more and more control over the outcomes of his work.

Karl's first projects, the rectangle TAM, and the set of repeated rectangles, CULL, were carefully planned and carried out. (Figure 10.1) His next design, ACE (Figure 10.2) was even more carefully structured, making use of symmetry and the properties of circles.

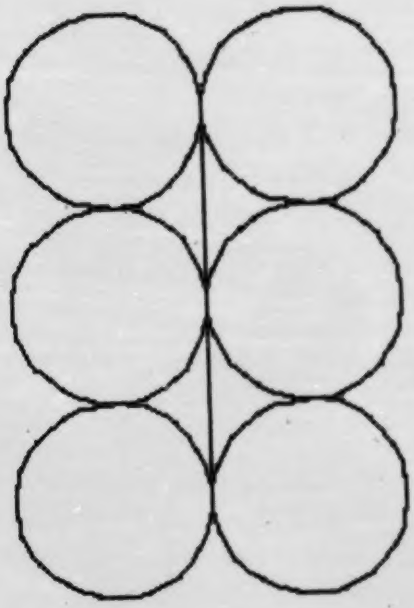
```
TO TAM
1 FORWARD 190
2 LEFT 90
3 FORWARD 100
4 LEFT 90
5 FORWARD 190
6 LEFT 90
7 FORWARD 100
END

TO CULL
1 TAM
2 TAM
3 TAM
4 TAM
END
```

Figure 10.1



```
TO ACE
1 RCIRCLE 50
2 LCIRCLE 50
3 FORWARD 100
4 BACK 200
5 RCIRCLE 50
6 LCIRCLE 50
7 FORWARD 100
8 FORWARD 100
9 RCIRCLE 50
10 LCIRCLE 50
END
```

Figure 10.2



Karl became fascinated by the effects of SPIN commands and began to add procedures together randomly to produce unexpected results. He would sit for long periods of time watching the effects of rapid SPINs interacting with various procedures. Each time he had a new idea for a project, he would add it to the others, to enjoy the wild unpredictable effects. Some of the procedures he created this way were:

```
TO ACE2          TO ME          TO ACE3
1 SPIN 1020      1 TAM          1 SPIN 1020
2 ACE            2 CULL         2 TAM
END              3 ACE          3 ACE
                 4 ACE2         4 ACE2
                 END            5 ME
                                6 HIDETURTLE
                                END
```

By the ninth class, these procedures culminated in:

```
TO BU
1 SPIN 200
2 CULL
3 ACE
4 ACE2
5 ME
6 NO
7 NO
8 XX78055
9 PLUS
10 TAM
END
```

The procedure NO was a sequence of slow SPINs and FORWARD commands, followed by HOME, while XX78055 and PLUS were carefully planned spinning designs created by using NO, rotating the TURTLE 10 degrees, and repeating the process while increasing the angle of rotation by 10 degrees each time.

```
TO NO
1 SPIN 100
2 FORWARD 100
3 SPIN 60
4 FORWARD 50
5 SPIN 70
6 FORWARD 60
7 HOME
END
```

```
TO XX78055
1 NO
2 RIGHT 10
3 NO
4 RIGHT 20
5 NO
6 RIGHT 30
7 NO
8 RIGHT 40
9 NO
10 RIGHT 50
11 NO
12 RIGHT 60
13 NO
14 RIGHT 70
15 NO
16 RIGHT 80
17 NO
18 RIGHT 90
19 NO
20 RIGHT 100
21 NO
22 RIGHT 110
23 NO
24 RIGHT 120
25 NO
26 RIGHT 130
END
```

```
TO PLUS
5 MAKE "BUR 10
10 NO
15 RIGHT :BUR
20 MAKE "BUR :BUR + 10
25 GO 10
END
```

A few classes after creating BU, Karl expressed the desire to control the effects of his procedures. In one period he edited BU to produce a very elegant spinning design, by systematically removing all the "random effects." Although he needed help with the mechanics of editing, the decisions about what to change were all his. The final version of the procedure was:

```
TO BU
1 SPIN 200
2 CULLEN
3 ACE
4 HOME
9 PLUS
END
```

The HOME command was introduced at line 4 to center the TURTLE before drawing the symmetrical spinning design, PLUS.

The editing of BU ended the "random phase" of Karl's work. Although he had periods of aimlessness, and fooling around with the keyboard, most of his actual work with the computer was carefully and purposefully planned in the classes that followed.

In addition to thoughtful geometrical planning (TAM, CULLEN, ACE, PLUS, etc.) and careful step-by-step debugging (as in his editing of BU), another process that Karl used to great advantage was finding the limits allowed by the computer in different areas.

For example, he used the process of successive approximations in converging to a limit, to find the largest input to SPIN. It took him nine tries, using successive inputs of 1000, 2000, 1050, 1001, 1010, 1040, 1020, 1030, 1025 before deciding that 1020 was the largest allowable input to SPIN. In doing this, he made use of the error message generated by the computer whenever he used an input that was too large.

In a similar fashion, he used the OUT OF BOUNDS error message and the method of successive approximation to find the boundaries of the display screen and the diagonal distance across it.

Karl also experimented with other limits of the computer system. He found the shortest procedure name (one letter) and the "longest" (his procedure QWERTYUIOPASDFGHJKLZXCVBN used every alphabetic symbol on the keyboard). He also found out what happened when you type very long numbers. (When he typed PR 900000000000000000000, the computer responded: 9.000002E31), or long strings of symbols. In general, he developed an experimental approach to the system -- "asking questions of it," by trying things, and seeing what happened.

Karl's major accomplishment was a set of procedures which he used to animate the TURTLE in such a way that he could "drive it around" on the screen. He was given the initial concepts for the procedures and filled in the specific instructions himself. The procedure ideas he was given were:

```
TO CAR                  TO CH
5 PENUP                 10 MAKE "LETTER KEY
10 MAKE "D 10           20 IF :LETTER = "R RT 30
20 CH                   30 IF :LETTER = "L LT 30
30 FD :D                END
40 GO 20
END
```

The command KEY was given as a "primitive" which "tells the computer which letter you type on the keyboard."

Using this basic idea, Karl was able to define a system of commands that made the activity interesting to him.

First he decided what commands he wanted, and what letters he could use for them. He decided to locate all the letters in one section of the keyboard to make them easier to find. He wanted to slow the TURTLE down, and to have it wrap around the screen without going out of bounds, so he added WRAP and WAIT 5 commands to his CAR procedure. He wanted to be able to decide whether or not to leave a trace, so he added PENDOWN and PENUP commands. After some discussion with the teacher, he decided that the speed of motion could be controlled by increasing or decreasing the size of each step, so he added commands to take care of that. Over a period of three or four classes he edited his procedures as follows:

```
TO CAR
1 WRAP
5 PENUP
10 MAKE "D 10
20 CH
25 WAIT 5
30 FORWARD :D
40 GO 20
END
```

```
TO CH
10 MAKE "LETTER KEY
20 IF :LETTER = "R RIGHT 30
30 IF :LETTER = "W LEFT 30
40 IF :LETTER = "F MAKE "D :D + 5
50 IF :LETTER = "S MAKE "D :D - 5
60 IF :LETTER = "A PENUP
70 IF :LETTER = "Q PENDOWN
80 IF :LETTER = "E MAKE "D 0
END
```

Notice that the letters "R" (right turn), "F" (faster), "S" (slower), and "E" (emergency stop), all are abbreviations for their functions while the letters "W", "A" and "Q" were chosen for their position on the keyboard.

Having created a very nice "toy" with the computer, Karl enjoyed playing with it a lot. He made some nice discoveries as he moved it playfully around the screen. For example, he found that repeatedly pressing "S" would eventually make the TURTLE go backwards at an accelerating rate. Or that he could draw a dotted line by alternating the keys "A" and "Q". He tried letting the car go very fast, drawing slanted lines wrapping around the screen, and experimented with the effect of many repeats of the same commands. At one point he realized that he could use this device to write his name in script -- but this was never fully completed.

Between periods of playing with his "car" Karl continued to pursue the idea of a racetrack, around which the car could be driven. He created the procedure TR which used circles to draw the racetrack, and included a starting and finishing line. In the last class he was beginning to work on a series of procedures that would allow the computer to keep track of the elapsed time for one circuit of the track. Although the project was not completed, Karl was able to define the logical structure of the race in a way that would have allowed a relatively easy translation to a set of computer procedures, had there been more time.

```
TO TR
10 LEFT 90
20 FORWARD 190
30 LEFT 90
40 LC 190
50 LEFT 90
60 FORWARD 100
70 RIGHT 90
80 LC 100
END
```

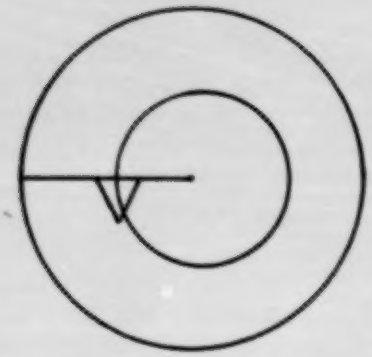


Figure 10.3

10.2 Major Difficulties Encountered by Karl in His Work

Karl's difficulties were related to his problems in reading and writing, based on hearing difficulties and a very poor visual memory. This was reflected in his painfully slow and poorly coordinated approach to using the keyboard. When Karl wanted to find a particular letter on the keyboard, he would scan the keyboard with his eyes, moving his finger back and forth, as his eyes shifted. Often his finger would pass the correct key several times before hitting it. If a wrong key was hit, an error message would result, and the whole process would begin again. Although Karl gradually improved his typing, his difficulty finding familiar keys continued to be noticeable.

Karl used the same random scanning strategy for finding numbers, as well. Although he knew that the numbers were on the top line of the keyboard, he was unable to make use of their inherent order to make them easier to find. This was particularly striking when he was numbering steps in a procedure. Going in a sequence, 1, 2, 3, 4, ... he conducted an individual "search" for each number, using his scanning and finger moving technique.

Karl also had great difficulty reading error messages. Although he gradually learned to include the new terminology in his sight vocabulary, he was unable to sound out words, even though he had seen them before. He either "knew" a word, or could not read it. Before long, he had become familiar with the most common error messages, and knew how to respond to them. He continued to have a problem whenever an unfamiliar error message appeared.

To help overcome these difficulties Karl made regular use of his notebook for writing the names of LOGO commands and of new procedures. He referred to these when he needed to recall the spelling or syntax of an instruction that he had already learned. When he had a list of commands on the screen that he wanted to copy into his notebook, he often asked either a teacher, or another student for help.

Karl also helped himself by limiting the size of his procedure names. He quickly realized that he could make procedure names very short; sometimes he wrote abbreviations for LOGO primitives (RC and LC for RCIRCLE and LCIRCLE, for example). Although he eventually used two and three letter names exclusively, he did not seem to choose his letters in a way that made scanning easier. Procedure names like TAM, ACE, CULL, and BU, all required a fair amount of scanning.

Karl named one of his procedures QWERTYUIOPASDFGHJKLZXCVBNM making use of every letter of the alphabet, in their keyboard sequence. Since he could move his finger in order across all the letters, this name required no scanning, and was actually a lot easier for him to type than a three letter procedure name for which he had to scan.

When designing his CAR procedures, he did make use of keyboard placement, in choosing the keys he used to control the motion of the TURTLE. The letters he used- - Q, W, E, R, A, S, and F -- are all located in a group on the left hand end of the keyboard. Originally Karl thought he would make a cardboard cover for the keyboard, with a hole cut in it so that only those letters could be seen. He found this to be unnecessary, however, and concentrating his attention on one small corner of the keyboard, was able to select the correct keys easily, without any of the scanning or memory problems that occurred when he had the entire keyboard as his field.

Given Karl's extreme difficulties with reading and with the keyboard, it seems significant that he persevered as consistently as he did in his LOGO work.

10.3 Affective Aspects of Karl's LOGO Experience.

At the beginning of the LOGO classes, Karl tended to have "deadpan" expressions at all times. This corresponded to his appearance when seen for interviews, or on random occasions throughout the school. Even when he was successful in using the computer for the first few sessions, his expression continued to be blank and non-committal.

As his success continued, Karl became more assertive and curious. He asked what error messages meant, and sought to understand how to use new commands. He made a point of finding out how to use the Plotter and Printer, so that he could make his own "hard copy" of his computer work. Observing this during one class, a regular observer wrote: "Karl seems more actively involved, engaged with, and using the computer, based on more of his own interests, curiosities, purposes, and playfulness. Today he inquired about the other equipment; and was interested in its functioning and products...Karl many times today is asking 'how' giving the impression of one interested in how things are done..."

At about the same time, Karl began to express an interest in the work of other children in his group, and began to show them his work. He invited a friend to class and swapped programs with him. He began to show that he was feeling good and enjoying himself. His face was more expressive, his posture more relaxed. He wrote his name in large block letters on a formica table which was used by the children as a graffiti board", and proudly began to show his work to visitors.

Karl brought his tutor from the school's learning center to one of the later classes, and showed her his partially completed "car" project, carefully explaining each detail of his programs to her. In the next class, the observer quoted above again noted changes in his behavior: "Karl entered the room with Jimmy and Darlene. He immediately set to logging in on terminal 4. I notice that he seems to walk taller with a more self-assured posture and has a way of getting right to the business at hand."

Changes in Karl's attitude toward his classroom work were also noted by his regular teacher. After about twenty LOGO classes she reported that Karl was beginning to show that he really cared about his school work, that he had begun concentrating on his work in a way that she had not seen before, and that he seemed to have a great deal more confidence in his ability to carry out academic tasks. She attributed these changes directly to his feeling of success in the LOGO classes.

Karl's success in his LOGO classes demonstrated that with an appropriate educational environment, he was able to function at a higher level of ability than he had demonstrated in schoolwork, even with a great deal of one-to-one tutoring. At the present time his seventh grade teachers and his learning disability tutors are trying to develop an educational plan for him that will capitalize on the abilities he has demonstrated in the areas of reasoning, logic, planning and organization, in order to help him maintain the feelings of success that he developed while working on LOGO.

In his final interview, after the end of the LOGO classes, Karl was significantly more articulate about intellectual activities (although not about his feelings). He listed ten uses for a tin can (as opposed to four uses for a brick in the first interview). Instead of carrying out the four color permutation task, he asked "Can I just show you how I do it?" and proceeded to describe a system for finding six permutations that started with each of four colors. With some difficulty, he correctly calculated that there were twenty four possibilities in all, and leaning over to speak directly into the interviewer's tape recorder, he said: "Twenty-four. I'm a brain!"

11. Kathy

Kathy, a student who was new to the school this year, is considered to be an "above average" student. (Her most recent school administered achievement tests place her in the 60th percentile overall.) She is cheerful, confident, and enjoys "playing" with words and ideas.

From the very beginning of the classes, Kathy displayed a quiet confidence, and competance in using the computer. Although not very assertive in the early group sessions, she seemed to have an excellent understanding of basics right from the start.

1. Kathy's Working Style

Kathy was a student with a subtle sense of humor who derived a great deal of intellectual pleasure from her work. She usually worked in an explorational mode, or in the context of short tasks, either self defined or suggested by a teacher. When difficulties were encountered, she preferred to resolve them on her own, although without a great deal of persistence. When she did ask for help, she was accepting of the teacher's suggestions, and readily learned new ideas in the context of the projects in which she was engaged.

Kathy carried out dozens of small projects in the course of her LOGO experience. She shifted back and forth between open ended explorations and small goal directed projects. Her favorite activity was to repeat and combine existing procedures to produce new unexpected results. Often she would interrupt an exploration to pursue a particular idea which had been suggested to her by the designs she had just created.

One of the ways in which Kathy structured her work was in her choice of procedure names. Her procedure names often indicated the relationship between a new procedure, and the subprocedures from which it had been built. Thus, a symmetrical design was called BARN because it was built by repeating a subprocedure called HORSE. A procedure called WORMY was made by doubling all the sizes in a similar design called WORM. And, in a rare example of top-down naming, a procedure called MONSTER was made up of subprocedures MO, NS, and TER.

Kathy was extremely comfortable with giving and receiving help. She often helped other children with the use of disks, particular elements of the LOGO language, ideas for projects, etc. She was also quite willing to ask for help when she needed it, both from the teacher and from classmates, especially Monica with whom she worked a good deal. In this way, Kathy was able to make steady progress in her understanding of LOGO, in her ability to conceive and carry out projects and in problem solving skills.

Kathy enjoyed working with concepts, rather than simply with practical results. She was the only student among the first eight to persevere in making the computer draw a circle. She spent a lot of time, talking with her teacher and playing TURTLE, until she understood that she could make a circle by repeating FORWARD 20, RIGHT 20, over and over again. She rejected the idea of accomplishing this with recursion (although it would have been quicker), because she was trying to follow through with her own idea of repeating each step, step by step. She did accept the idea to combine several of the steps into a subprocedure, and then repeat the subprocedure to make a circle. When her final result, SHELL, eventually had too many steps, and went on past the closing point of the circle, she was satisfied with it. She knew that she could have modified her circle to make it close exactly. What she had been concerned with was whether the circle would close at all. When it did, she was satisfied that she had solved her problem.

```
TO ROUND
1 FORWARD 20
2 RIGHT 20
3 FORWARD 20
4 RIGHT 20
5 FORWARD 20
6 RIGHT 20
7 FORWARD 20
8 RIGHT 20
9 FORWARD 20
10 RIGHT 20
END
```

```
TO SHELL
1 ROUND
2 ROUND
3 ROUND
4 ROUND
END
```

This was typical of Kathy's approach. She developed a concept of what she wanted to do. She asked for help when she felt she needed it. She listened to the various suggestions, and selected from them the ones she wanted to follow, in accordance with her own understanding of what the problem was about. Whatever approaches she used in her problem solving, she learned. While she often asked for help, she did not need to ask for help in those areas again.

Kathy was comfortable initating ideas for projects, and borrowing them from others -- even copying procedures directly from a booklet or bulletin board. She quickly learned that direct copying often led to unexpected problems, and she became more careful with her borrowing.

Kathy and Monica worked together a great deal. (See profile #14, Monica.) Both of them were interested in small, short-term projects, with visually pleasing results. Both depended on their mutual sharing as a source of ideas, help and reassurance. Although they often worked on the same tasks, they usually worked separately. Their approaches, and results, were different. Kathy's favorite activity was making a procedure and repeating it. Monica's was repeating a procedure and putting a rotation after each repeat. For example, Kathy borrowed a procedure of Marilyn's called HORSE in which the computer repeated the instructions, BOX, RIGHT 20, five times. Kathy changed and elaborated it as follows:

```
TO HORSE
1 BOX
2 RIGHT 70
3 BOX
4 RIGHT 70
5 BOX
6 RIGHT 70
7 BOX
8 RIGHT 70
END
```

and she repeated it, using:

```
TO BARN
1 HORSE
2 HORSE
3 HORSE
4 HORSE
5 HORSE
END
```

Kathy made BARN, after repeating HORSE several times by direct command and deciding that she wanted to repeat HORSE exactly 5 times. Although she could have used recursion, she chose to limit her repeats to exactly five. She also called her new procedure BARN, introducing the mnemonic device "a barn is a group of horses." Once again, Kathy was "in charge" of what happened, using the cliche idea, "repeat a procedure over and over," keeping control of both the process and the end result.

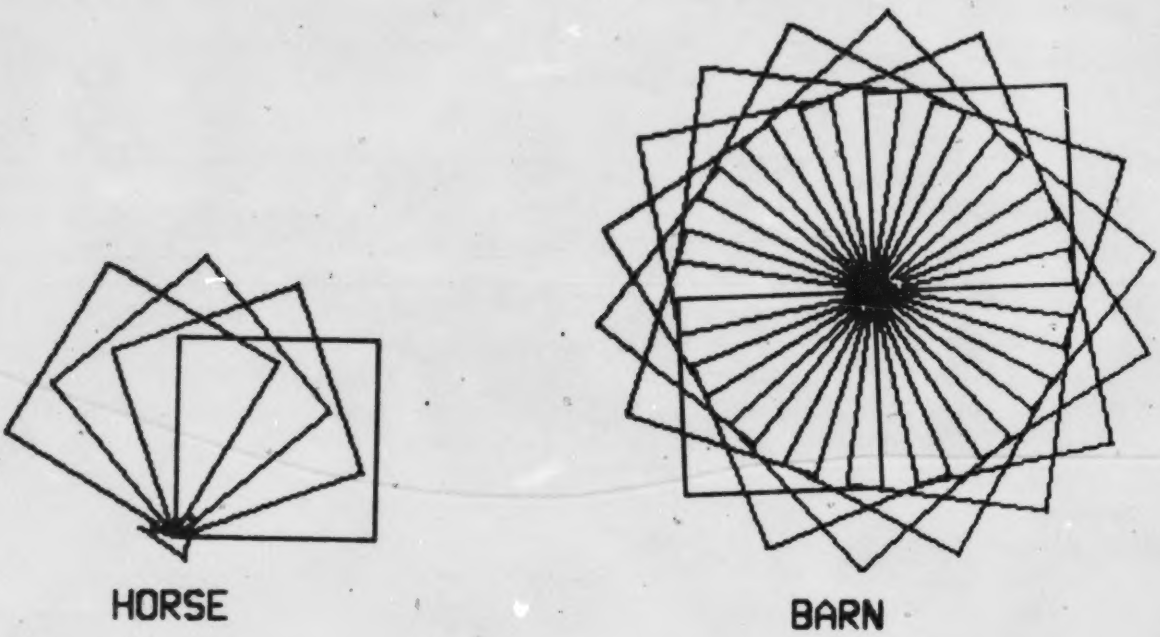


Figure 11.1

Thus when Kathy and Monica worked on similar tasks, each child brought her own specific knowledge, experience and "bag of tricks" to the project; and came out with a result that made her feel successful. Looked at in detail, we see that the girls actually functioned quite differently, and what each of them learned from the project was probably quite different. There does not seem to have been any great significance to the choice of angle (20 degrees for Monica and 70 degrees for Kathy). Both were simply using a number that had worked out well before. Kathy's 70 may have been just a mis-copying of Monica's 20.

2. Some Examples of Kathy's Work

Kathy's approach to her work is exemplified by a series of small projects which made use of a BOX and a TRIANGLE procedure as fundamental building blocks. The BOX and TRIANGLE procedures were constructed during periods of careful, goal-directed explorations.

```
TO BOX
1 FORWARD 100
2 RIGHT 90
3 FORWARD 100
4 RIGHT 90
5 FORWARD 100
6 RIGHT 90
7 FORWARD 100
END
```

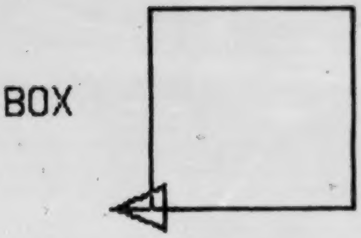


Figure 11.2

```
TO TRIANGLE
1 LEFT 90
2 FORWARD 100
3 RIGHT 120
4 FORWARD 100
5 RIGHT 120
6 FORWARD 100
END
```

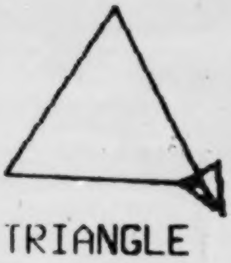


Figure 11.3

Box was the first procedure completed by Kathy and her group, and they immediately followed by constructing a series of figures making use of BOX as a subprocedure. (See examples in Part II, Chapter 5. section 1.3.2). It was quite natural for her to repeat TRIANGLE as well. She was pleased with the result, calling it BUTTERFLY. She then repeated BUTTERFLY six times until the figure "closed". This new design she called 7BUTTERFLY (reflecting an initial miscount of how many repeats of the BUTTERFLY procedure she had used).

```
TO BUTTERFLY
1 TRIANGLE
2 TRIANGLE
END
```

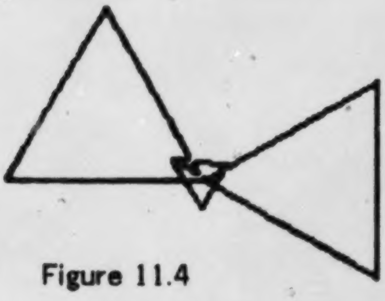


Figure 11.4

```
TO 7BUTTERFLY
1 BUTTERFLY
2 BUTTERFLY
3 BUTTERFLY
4 BUTTERFLY
5 BUTTERFLY
6 BUTTERFLY
END
```

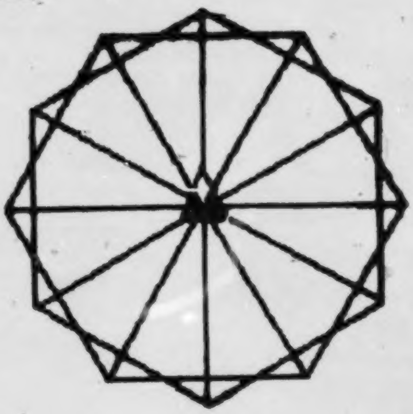


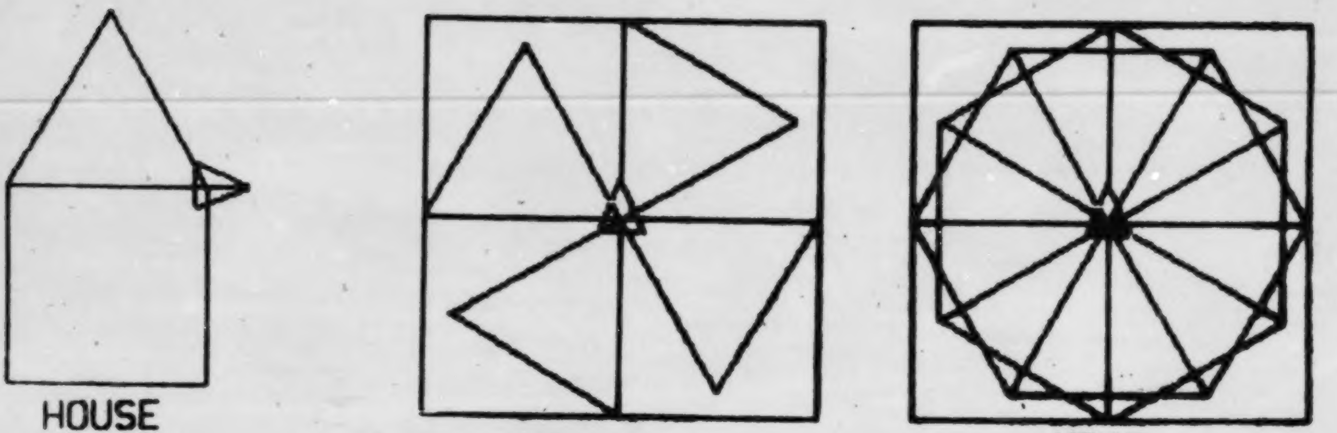
Figure 11.5

Following her initial exploration with triangles, Kathy's teacher suggested that she put her TRIANGLE and BOX procedures together to make a "house". After some goal-directed exploration, the HOUSE procedure resulted. Kathy immediately repeated HOUSE four times (calling this new procedure HOUSE4) until the design closed. Next she wanted to see how her HOUSE4 design and 7BUTTERFLY would go together. She named the result HB47, indicating its relationship to HOUSE4 and 7BUTTERFLY.

```
TO HOUSE
1 TRIANGLE
2 RIGHT 30
3 BOX
END
```

```
TO HOUSE4
1 HOUSE
2 HOUSE
3 HOUSE
4 HOUSE
END
```

```
TO HB47
1 HOUSE4
2 7BUTTERFLY
END
```



HOUSE HOUSE4 HB47

Figure 11.6

This set of projects culminated when Kathy declared that HB47 "looks like a spider," and returned to goal-directed activity, adding a series of circles to the design, to produce the procedure SPI.

```
TO SPI
2 RCIRCLE 30
3 LCIRCLE 30
4 RCIRCLE 20
5 LCIRCLE 20
6 BACK 30
7 RCIRCLE 10
8 LCIRCLE 10
END
```

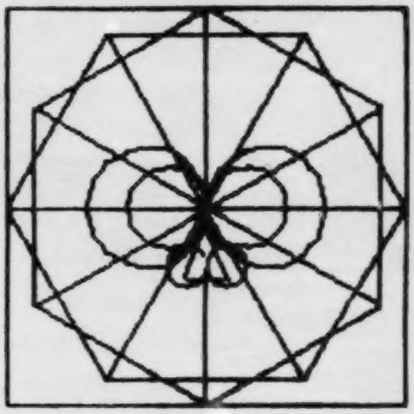
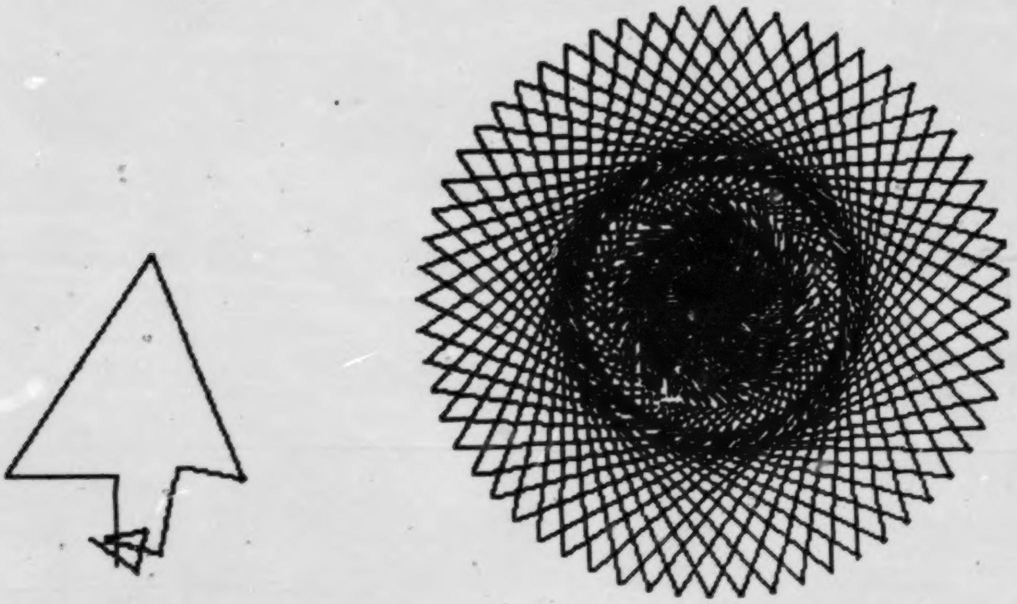


Figure 11.7

Although Kathy had constructed HB47 and its subprocedures by repeating simple shapes over and over to make a symmetrical design, she was also able to make explicit use of both right/left symmetry and similarity of shape in the process of constructing her "spider". It was this combination of more or less random explorations involving existing procedures, with expert use of heuristics such as similarity and symmetry when working in a goal-directed manner, that most commonly characterized Kathy's work.

When Kathy repeated her borrowed XMASTREE procedure, she found that many repeats made a lovely, complex design. Here she used recursion, since she was not concerned about how many times the procedure was repeated in all. (Figure 11.8)



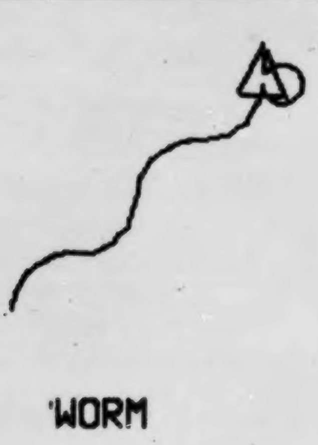
XMASTREE

STAR

Figure 11.8

Later in the series of classes Kathy did a lot of exploration with arcs and circles, and began to work on some longer projects. One day she decided that a series of arcs "looks like a worm," and wrote the procedures WORM and WORMY (in which WORMY is exactly twice the size of WORM). (Figure 11.9)

```
TO WORM
1 RARC 30
2 LARC 30
3 RARC 30
4 LARC 30
5 RCIRCLE 10
END

TO WORMY
1 RARC 60
2 LARC 60
3 RARC 60
R LARC 60
5 LCIRCLE 20
END
```



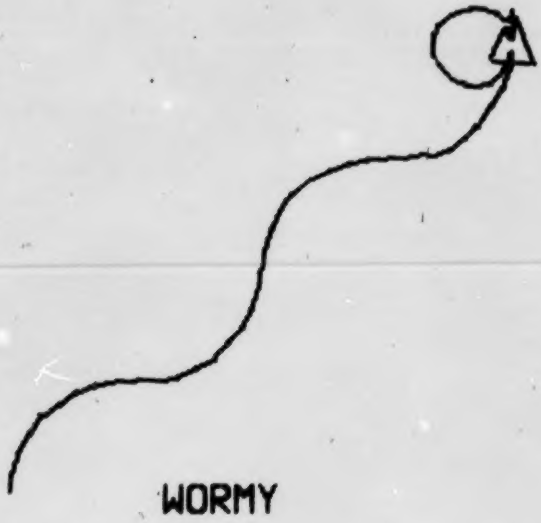


Figure 11.9

In a later experiment, which made use of symmetry in an unusual way, Kathy created an exotic looking "Monster." (See Figure 11.10).

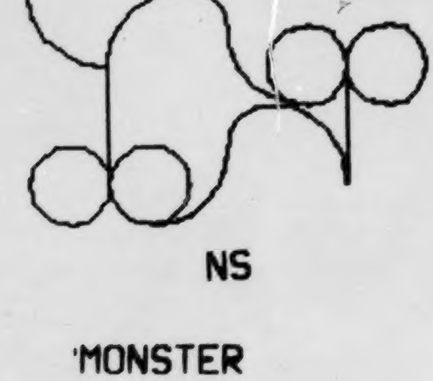


Figure 11.10

When she decided to teach MONSTER to the computer, the teacher suggested that she break up the project into parts. She isolated three parts, and decided to name them MO, NS, and TER, so that her procedure MONSTER would be:

```
TO MONSTER
1 MO
2 NS
3 TER
END
```

and the subprocedures were:

```
TO MO
1 RARC 40
2 RARC 20
3 LARC 40
4 LARC 20
5 LCIRCLE 20
6 RCIRCLE 20
END
```

```
TO NS
1 LARC 40
2 LARC 20
3 RARC 40
4 RARC 20
5 RCIRCLE 20
6 LCIRCLE 20
END
```

```
TO TER
1 RARC 40
2 RARC 20
3 LARC 40
4 LARC 20
5 RCIRCLE 20
6 LCIRCLE 20
END
```

Kathy may have realized that MO and TER were identical, but in any case, she wanted to have TER as a distinct procedure to follow through on her naming scheme. Now, when Kathy typed MONSTER, the result was.

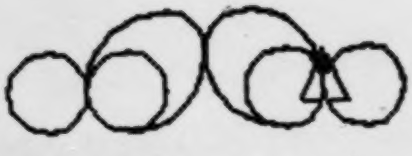


Figure 11.11

Kathy had forgotten to include the interface steps between the three procedures, so that when she ran the procedure MONSTER, the result was quite discouraging. When asked for help in debugging the teacher suggested running MO, NS and TER separately. When Kathy did so, she could see that the intermediate steps had been left out. Her teacher helped her to work out what these steps should be, and in its final incarnation, MONSTER became:

```
TO MONSTER
1 MO
2 BACK 60
3 NS
4 FORWARD 60
5 LEFT 90
6 TER
7 LARC 40
END
```

Kathy's last project, carried out during classes 21 and 22, also involved symmetrical arcs, and also required a good deal of debugging. In this case, Kathy had tried out a long sequence of direct commands, and made a mistake or two in copying them into her notebook. When her procedure turned out to be buggy, she had to spend a lot of time stepping through it, in order to figure out which steps were wrong, and how to fix them. Since she had numbered all her steps by ones, she had to do a great deal of unnecessary retyping. At several points in her work it had been suggested to Kathy that she number steps by fives or tens, but she had never felt a need to adopt that suggestion. (In the class following this one, Kathy had a visitor, Renee, who was learning to write a procedure. When Kathy taught Renee, she told her to number the steps by tens. It seems that she got the point, however belatedly.)

3. Exploring the Effects of Different Angles

If Kathy's work had an area of weakness, it was in Turtle Geometry. Kathy had difficulty estimating angles, and tended to stay away from projects that made it necessary for her to work precisely with angle manipulations. Although she and the teacher had "talked through" the idea that "when the turtle goes all the way around, it turns 360 degrees," as part of Kathy's circle project, this became an idea that she "filed" away, and did not find much use for on her own.

To help Kathy focus on the effects of using different angles, she was shown a POLY :ANGLE procedure. While she liked the designs that it made, she did not analyze the connection between the input number, and the shape that resulted. In her first session experimenting with POLY, she used the inputs 88, 234, 12345, 300, 344, 90, 199, 125, and 888, 666, 555, and 77. Although she used POLY again on several other occasions, she continued to choose inputs fairly randomly.

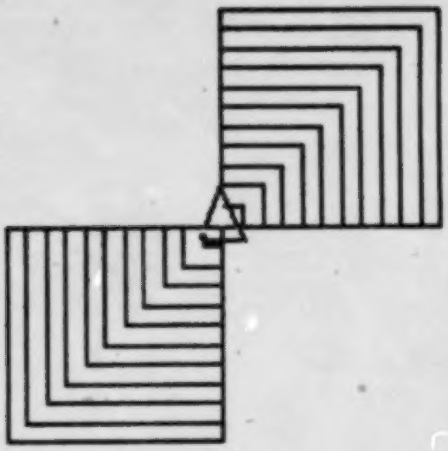


Figure 11.12

During class 20, she was experimenting with her procedure WOW which drew a set of nested squares. She was trying to rotate this to create a particular design, (Figure 11.7) but could not determine the correct angle despite several attempts. To help her focus on the problem, she was shown the procedure.

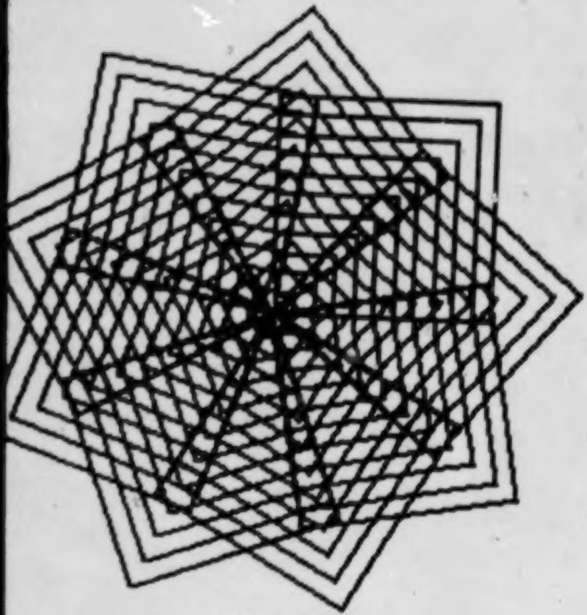
```
TO SPINWOW :ANGLE
10 WOW
20 RIGHT :ANGLE
30 IF HEADING = 0 STOP
40 SPINWOW :ANGLE
END
```

This was meant to serve as a vehicle for exploring angles, and for furthering her understanding of variables and STOP rules.

At first she chose inputs like 900, 9999, 777 and 666. At the beginning of the next class, a discussion was held about "interesting angles," and she was reminded that the TURTLE turns all the way around in 360 degrees. It was suggested that numbers that divided evenly into 360 degrees might be "interesting numbers" and that she spent the period experimenting with SPINWOW, and taking notes on the results.

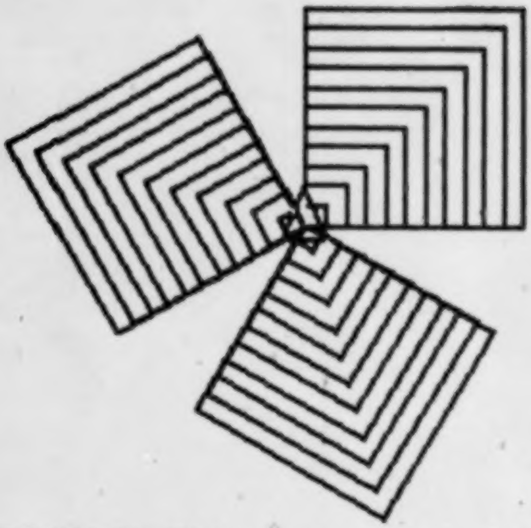
Initially Kathy began using inputs that were factors of 360, like 4, 12, and 18, 60 and 90. She quickly branched out to 100, 200, 400, etc. Since she had been specifically asked to take notes on the results, she paid careful attention to what was happening, for the first time. For example, she counted the number of "cones" that appeared in the designs, and this showed her that certain figures (SPINWOW

Figure 11.13

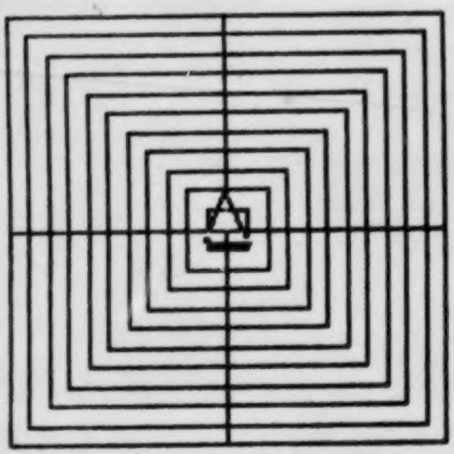


SPINWOW 40

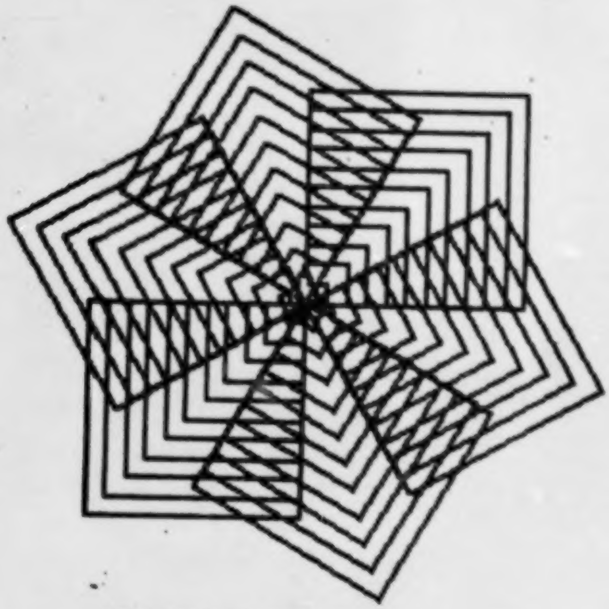
PINWOW 200 SPINWOW 400



SPINWOW 600 SPINWOW 120



SPINWOW 90



SPINWOW 60

200 and SPINWOW 400) looked "the same". Her notes, entitled "Interview with SPINWOW'S" are copied from her notebook (see Figure 11.13). (compare this with Monica's work with her procedure WISHWOW).

Interview with SPINWOWS

SPINWOW 40: it had 9 points it looked like a spiders web.
SPINWOW 200: looks exactly like spinwow 40.
SPINWOW 400: looks exactly like the two above.
SPINWOW 600: it has three points looks like a martian face.
SPINWOW 120: looks like SPINWOW 600 has that martian face look.
SPINWOW 30: it has 12 points looks like a combined thing of a snowflake and a spiders web.
SPINWOW 90: it just makes a biger wow
SPINWOW 140: it has about 17 points looks like a snowflake.
SPINWOW 60: looks like a wow that was done 6 times has 6 points.

. . . .

Kathy liked to initiate new projects, was comfortable with new ideas, and enjoyed the challenge of working on something to which the answer was not known in advance. She accepted the existence of bugs - even coined the phrase "exterminating" to replace "debugging" -- and was willing to work to resolve them. On the other hand, she usually chose to work on small projects, and to carefully limit the tools required for any task that she set herself. This was her way of remaining in control of her work, making sure that her experience was not too confusing for her.

If Kathy continued with LOGO classes, she could have worked on more projects using angles as variables, and on at least one long term project that required advanced planning and the use of subprocedures. More work in these areas would have nicely rounded out her LOGO experience.

12. Kevin

Kevin is a conscientious student who is considered to be "below average" in most of his school work. On his sixth grade achievement tests his overall national percentile ranking was 45. Nevertheless, Kevin was consistently a very able student in working with LOGO.

Kevin began the series of classes with a confident and accurate control of the TURTLE, which persisted throughout his work. He did not initially have the same sureness in using the computer as a tool to simplify and organize his work. Kevin's most significant project was the design and animation of a large "turtle", which he drew on the display screen using circle and arc procedures. While working on this project, he began to use the idea of subprocedures and state transparent procedures to simplify his work. During the last few classes he worked on projects involving the use of two and three variables to produce designs (such as his TUNNEL procedure, Figure 12.11) which used the idea of similarity as a guiding principle.

Kevin demonstrated a clear understanding of the concept of variables and was able to add variables to his procedures to control both the size and shape of the design elements and the starting and stopping of the procedure. He progressed in his work from using the computer to control the TURTLE, to learning how to control the processes of the computer itself.

1. Kevin's Working Style

In the first few classes Kevin demonstrated an exceptional ability to control the TURTLE. He was quick with accurate estimations and changed or combined steps with confidence. Kevin was the first student in his group to see that a rotation of 90 degrees was necessary to make a square, and the first to see that RIGHT 90, RIGHT 90, could be replaced by one instruction, RIGHT 180, to turn the TURTLE around. He rapidly learned to shift the position of his head, to parallel the position of the TURTLE, in order to decide which direction to turn it. In general, the world of the TURTLE was a very comfortable one for him.

Kevin was a compulsive note taker. He took notes as he worked, and then recopied them, so that they would be "neat" in his notebook. When the children all worked as a group for the first four classes, Kevin was the first to take on the role of "recorder". At the end of the first day, after the group had made the TURTLE draw a box, but had not yet learned how to write a procedure, Kevin wrote down the steps in his notebook, so that they could be remembered:

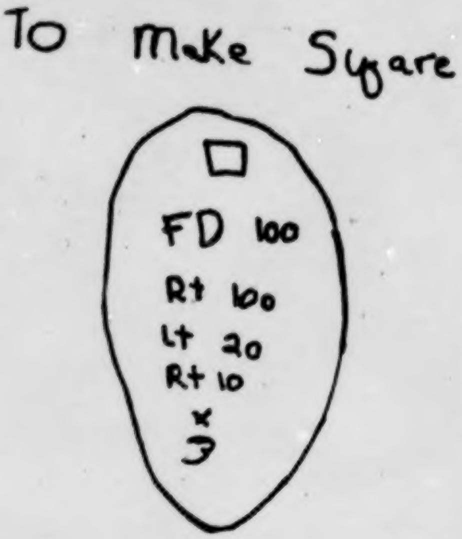


Figure 12.1

With the exception of one project, Kevin made a concerted effort to finish everything he started. He was willing to experiment patiently, and when something did not work out exactly right, start again. His careful notes of successful steps, enabled him to start easily from where he had left off.

Kevin's major difficulty in working with the computer was an initial reluctance to plan ahead, or to think about and structure his work more than one step at a time. The teaching strategy used to deal with this was to supply Kevin with new ideas, at the moment when they made the greatest sense to him. When they simplified his work or answered an immediate need. In this way he was able to assimilate new ideas, and incorporate them in his subsequent work.

Kevin was very responsive to suggestions from the teacher. He absorbed new ideas quickly, when they were relevant to his work and to his existing sense of how things could be done. In the same way that Kevin automatically combined steps to rewrite FORWARD 50, FORWARD 50, as FORWARD 100, the next time he wrote it, or RIGHT 90, RIGHT 45 as RIGHT 135, he readily sought out and accepted other ideas that led to shortcuts, or streamlining in a particular project.

Kevin had a linear approach to problem solving. He approached tasks one step at a time, in order. Advance planning was not his specialty. Although he did learn to use subprocedures within a larger project, when he had one part that was repeated over and over, or when he needed to break up his work into manageable chunks, he was never interested in "top down" planning, in which he would have had to decide in advance which subprocedures he wanted to use, and how to organize them.

Kevin's work during the series of classes can be divided into a few major segments. During the first four sessions he worked with the entire group, learning the basics of LOGO by developing some designs involving squares. Next he invented a triangle procedure, and created other shapes using the triangle. After abandoning a complex flag project he began his largest project -- making the computer draw and then animate a large "turtle". He worked on this for the major part of six different classes. During his last three working sessions Kevin worked with POLY procedures, exploring angles, shapes, variables and stop rules.

During the first four classes, Kevin took a lot of leadership, as the class worked together on a series of projects involving boxes. Kevin suggested ideas for projects, names for projects, and was especially helpful in suggesting the steps needed to carry out the projects. It was only when the teacher suggested that one of the projects be carried out in a top down, plan-ahead manner, that Kevin had difficulty understanding what was happening, and how to proceed.

2. Kevin's Early Projects

Kevin's intuitive grasp of Turtle Geometry became more apparent during his first independent project -- making a triangle. His very first attempt was RIGHT 45, FORWARD 100, RIGHT 45, RIGHT 45, FORWARD 100, producing the figure shown.

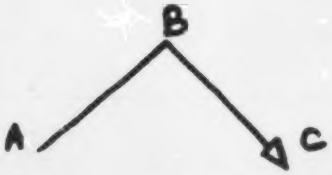


figure 12.2

He needed two tries, experimenting with the angle at point C, before hitting on RIGHT 45, RIGHT 40, RIGHT 50, which he first combined to RIGHT 45, RIGHT 90, and then to RIGHT 135, when he wrote the procedure. In estimating the distance to close the triangle he first chose FORWARD 150, then moved the TURTLE BACK 10, and changed the total to FORWARD 140 in his procedure. Kevin's triangle procedure, arrived at in about 25 minutes of exploration was:

```
TO OF
1 RIGHT 45
2 FORWARD 100
3 RIGHT 90
4 FORWARD 100
5 RIGHT 135
6 FORWARD 140
END
```

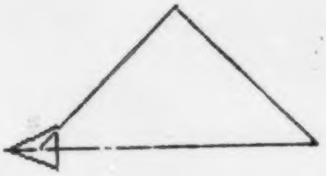


Figure 12.3

When Kevin repeated OF, it made a flower-like design. He also used his OF procedure, along with BOX, to make his own version of a house, and a row of two houses.

```
TO FLOWER
1 OF
2 OF
3 OF
4 OF
5 RIGHT 180
6 FORWARD 200
END
```

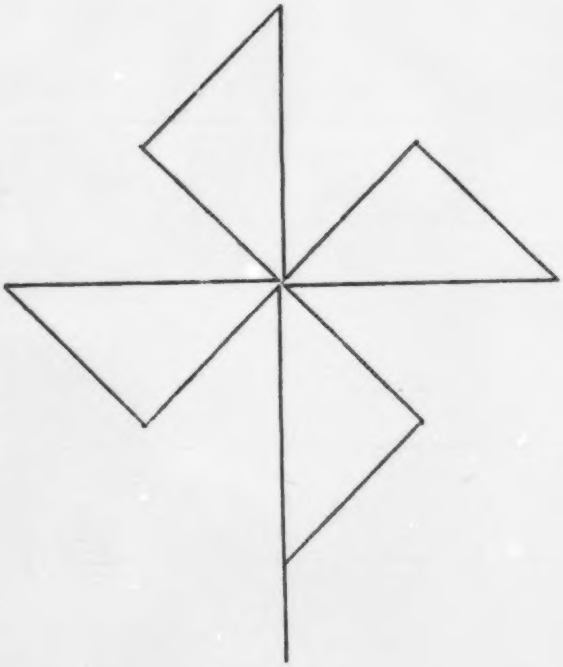


Figure 12.4

```
TO HOUSE
1 BOX
2 RIGHT 90
3 FORWARD 100
4 LEFT 90
5 FORWARD 20
6 RIGHT 90
7 OF
END
```

```
TO 2HOUSE
1 HOUSE
2 PENUP
3 FORWARD 100
4 RIGHT 90
5 LEFT 180
6 FORWARD 100
7 RIGHT 90
8 FORWARD 50
9 RIGHT 90
10 PENDOWN
11 HOUSE
END
```

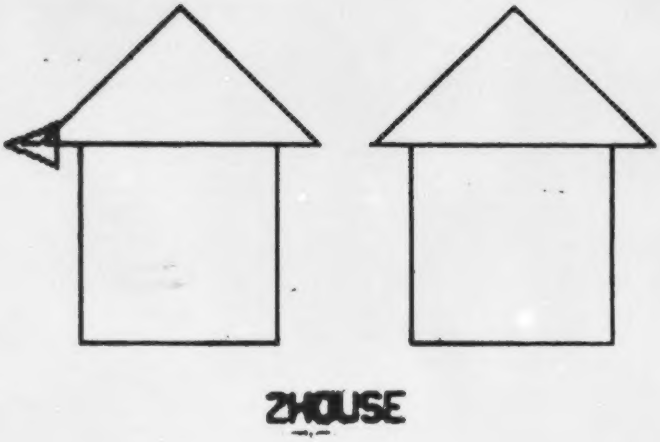


Figure 12.5

Kevin's next project was to make a large American Flag. His flag was worked out as a long sequence of steps. It was based on the procedure BIGBOX, which provided a background of 100 by 100 squares in a 4 by 3 grid. Kevin used the boxes of the grid as markers, as a kind of "coordinate system," so that he could tell how long the stripes should be. The flag, which Kevin worked out experimentally by direct command, looked something like this:

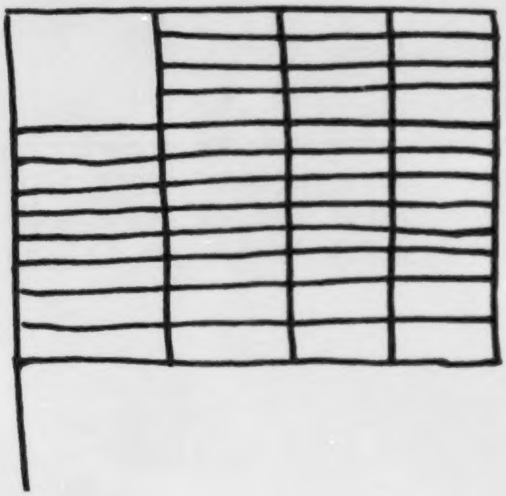


Figure 12.6

Kevin took careful notes, and attempted to write down each successful step in his notebook. Unfortunately, there were many steps, and Kevin made a few mistakes in copying. The process of debugging the incorrect steps, and restoring correct ones seemed too formidable to Kevin, and he decided to abandon the project. The teacher used the situation to suggest a different approach: plan out the project, using subprocedures for the long and short stripes, etc., Somehow, Kevin did not understand this approach. Or he did not want to "shift gears" and start over. He preferred to drop the project -- the only time Kevin ever gave up on anything he started.

3. Kevin's "Turtle" Project

Next, Kevin began what became his major project -- lasting for six classes, over a two week period. This time, he did a certain amount of advance planning. First he drew a picture of what he wanted the "turtle" to look like (Figure 12.7), and started right in to construct it, using arc and circle commands. This time, Kevin was willing to work in terms of subprocedures: SHELL, HEAD and FOOT (at the teacher's suggestion), but his approach to building the "turtle" was still sequential -- a step by step process.

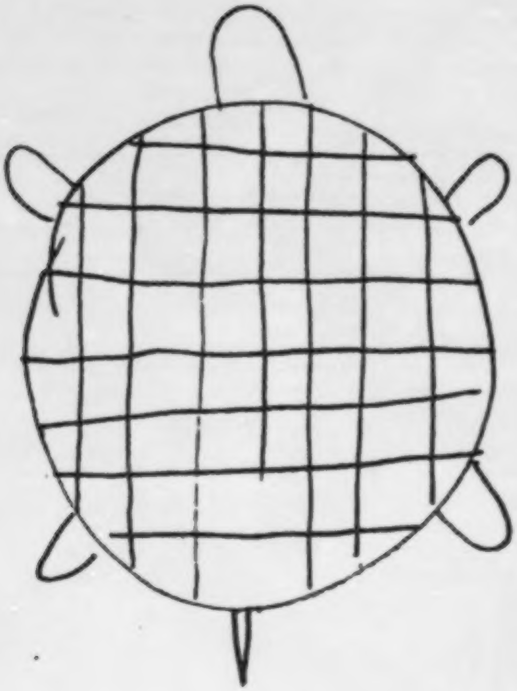


Figure 12.7

While working on this project, Kevin absorbed a lot of new techniques. He became expert in the use of PENUP mode for explorations. He learned to use RARC and LARC with precision, carrying out careful explorations with the TURTLE's pen up by using, for example, RARC 90, followed by RIGHT 180, LARC 90, RIGHT 180. Kevin was also shown an arc procedure with a variable angle, ARCR :R :A and ARCL :R :A, so that he could easily move the TURTLE around the circle which formed his "turtle's" outer shell.

As he had done for the flag project, Kevin invented his own coordinate system to aid him in moving around the outer shell: the 3500 display system leaves a dot after each individual step. The RCIRCLE procedure Kevin was using made use of angles of 10 degrees as the basic step. By experimentation, he discovered that ARCR 90 60, for example, would move the TURTLE exactly six dots along the outer circle. In this way, Kevin was able to navigate the TURTLE around the SHELL in a precise manner, to locate the "turtle's" feet, and tail, in relation to its head and each other.

Another device which Kevin used to draw his "turtle", was the use of two modular subprocedures, FOOT and BKFOOT. Kevin would move the TURTLE around the "turtle's" shell using an arc command until he reached the point where he needed to locate a foot of the "turtle". With the help of his teacher, he wrote the procedures FOOT and BKFOOT. These were almost equivalent to a state transparent procedure which drew a foot of the "turtle" and returned the TURTLE to the shell, ready to move around to the next point at which a foot was needed.

```
TO FOOT                   TO BKFOOT
5 FORWARD 8               10 FORWARD 11
10 RARC 20                20 LARC 20
15 RARC 20                30 LARC 20
20 FORWARD 11             40 FORWARD 8
END                       END
```

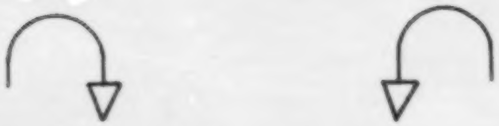
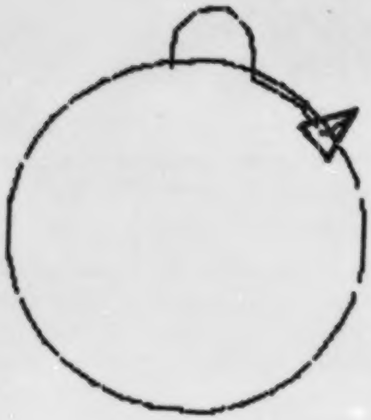


Figure 12.8

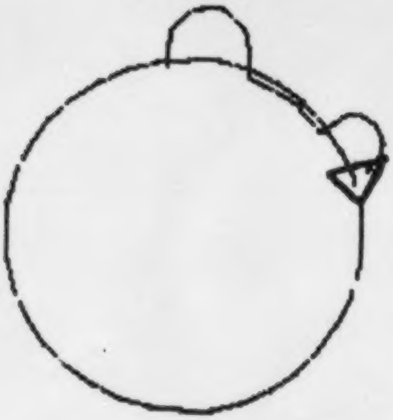
In this way, Kevin could move along the circle until he came to the point where a FOOT was wanted. He would then use the following sequence: LEFT 90, FOOT, RIGHT 180, BKFOOT, LEFT 90. This would return the TURTLE to exactly where it had been before making the FOOT, ready to continue its journey around the SHELL (Figure 12.9).



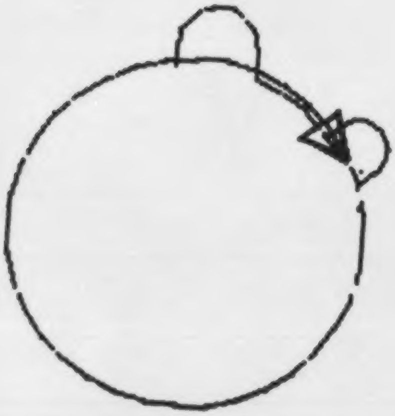
SHELL

HEAD LT 70 ARCR 50 30

LT 90



FOOT RT 180



BKFOOT LT 90

Figure 12.9

This example is discussed in detail, because it gives a good sense of how well Kevin absorbed new ideas, at the moment that he needed them, as long as they fit into the basic approach he was using. He rejected the idea of making the FOOT procedure itself state transparent (the teacher's idea), but combined it with BKFOOT in his own way for exactly the same purpose. Once he adopted the idea, he used it expertly, without hesitation. It was then fairly easy for him to complete the "turtle" (Figure 12.10). He had abandoned the idea of making lines across the shell.

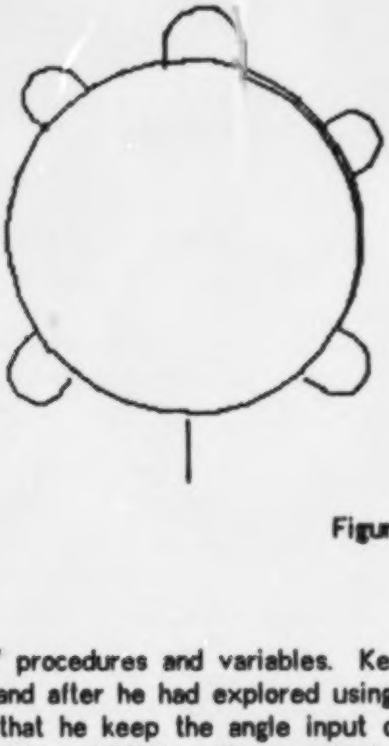


Figure 12.10

4. Kevin's Work with Variables

Kevin's last few projects involved POLY procedures and variables. Kevin was shown a POLY :SIDE :ANGLE procedure, and after he had explored using it with different inputs, the teacher suggested that he keep the angle input constant, while varying the size. He then drew a series of POLYs with a constant angle of 45 degrees. POLY 50 45; POLY 55 45...POLY 110 45. This made a design that Kevin called a "tunnel". He and the teacher talked through the procedure

```
TO TUNNEL :SIZE
10 POLY :SIZE 45
20 IF :SIZE = 105 STOP
30 TUNNEL :SIZE + 5
END
```

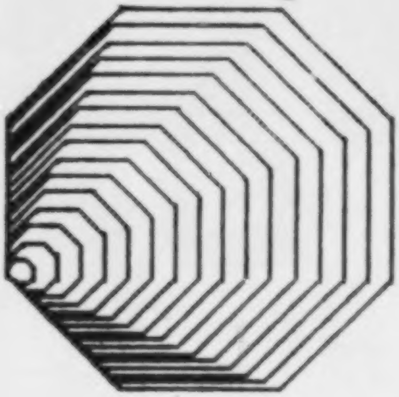
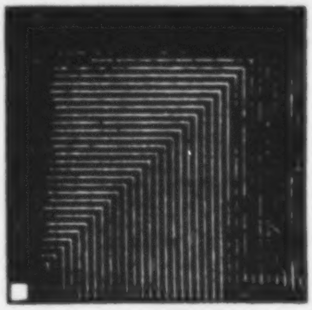


Figure 12.11

During the next class, Kevin made a great leap in understanding the use of variables. He experimented with varying the size of a POLY whose angle was 90. Then, with some help from the teacher, he wrote the procedure:

```
TO LIFS :SIZE
10 POLY :SIZE 90
20 IF :SIZE =150 STOP
30 LIFS :SIZE + 2
END
```



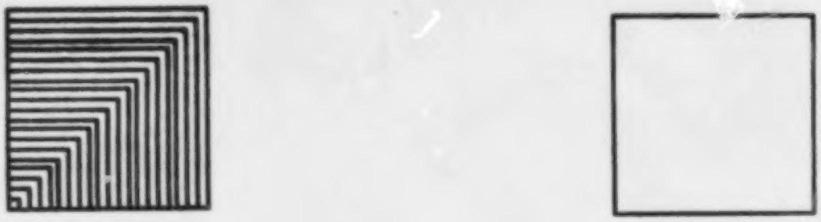
LIFS 10

Figure 12.12

Kevin then asked if "the amount the POLYs grow each time" could be changed, and if the largest size could be changed. He picked the variable names SET and LARGE for these quantities, and with some syntax help from his teacher, wrote the procedure:

```
TO UFC :SIZE :SET :LARGE
10 POLY :SIZE 90
20 IF :SIZE = :LARGE STOP
30 UFC :SIZE + :SET :SET :LARGE
END
```

In the course of exploring the use of different inputs to this procedure, Kevin was delighted to discover that keeping all the inputs the same had the effect of producing a variable sized square. He understood that the reason the procedure drew only one square was that the starting and ending sizes were identical.



UFC 5 5 100

UFC 100 100 100

Figure 12.13

During these sessions, Kevin demonstrated the same quick learning in the use of variables, that he had shown with regard to Turtle Geometry. This was possible because he was using these elements, manipulating them to achieve a purpose. Always ready to absorb an easier way of doing something, Kevin was able to make a leap in understanding that he had not achieved earlier, when the idea of variables and stop rules had been introduced as a lesson for the whole group.

5. Conclusions

Kevin began the series of classes with a very strong and accurate sense of how to control the TURTLE. He did not originally show the same sureness in using the computer as a tool to organize and simplify his work. While working on his "turtle" project, he began to use subprocedures, and state transparent designs to simplify his work. By the end of the series of classes he had assimilated the idea of using variables to control the size and shape of repeated POLY designs, and to control the procedure itself. Thus he had moved in his work from using the computer to control the TURTLE, to learning to control the computer itself.

13. Laura

Laura is considered to be an average student by her teachers. (On her most recently recorded school achievement tests, however, her national percentile ranking was 73.) Laura got off to a good start in her LOGO work, quickly mastering basic TURTLE commands, and the use of procedures and subprocedures. By session 8 she had completed a substantial project -- drawing a face using a top-down program structure with subprocedures.

She worked really well for three or four more sessions -- after which her progress bogged down a bit as several new ideas were introduced in close succession, and the projects she was trying to do became harder.

1. Laura's Working Style

Laura had a tendency to try to learn quickly. She wanted to "know" the answer immediately and preferred not to ask for help except when absolutely stuck. In addition, she had a difficult time articulating her purposes in working. Whether this was due to the fact that she herself did not know her purposes, whether she knew them but could not articulate them, or whether she just felt that they were "private" and did not want to share, is not entirely clear. What is clear, according to an analysis of her dribble files, is that there were times when she was confused about left and right turns, about the effects of a series of steps in a procedure, or about error messages generated by misunderstandings about LOGO syntax -- and she did not ask for help, or use any other thoughtful strategies for clarifying her confusion. Laura often evidenced confusion by appearing bored, or by acting in a particularly "perky" manner. In describing Laura's experience we will try to identify what we believe she learned solidly, and what she evidenced confusion about.

Laura showed great interest in making large, freely conceived designs on the display screen. She created the designs one step at a time, considering thoughtfully the size and placement of each new addition to her creation. It was difficult for Laura to make the transition to formalization of her work; to breaking it down into small tasks; and to planning and organization. Consequently, there was often a gap between what Laura wanted to accomplish, and what she was able to accomplish. Laura did carry out a few major projects: a FACE project with several subprocedures; a series of designs using circles and squares of variable sizes constructed by means of recursive procedures with changing inputs; a "madlib" language game for which Laura created the basic story, wrote out lists of nouns, verbs, adjectives, and adverbs, and for which the teacher helped with most of the programming; and causing the computer to draw her initials.

Sometimes Laura appeared to be bored. In hindsight, this appears to have been a manifestation of her confusion. Too much stress was placed on offering her new ideas, rather than understanding her confusion, and taking steps to help her limit her choices and consolidate her earlier learning. Laura's difficulties were compounded by the fact that she did not like to ask for help, she did not like to be observed in her work, and she assumed an "air of confidence" at all times.

At the very first session, Laura showed great interest in procedure names and in "communicating" with the computer. She noticed "FOO" on the display screen, and asked "Does the computer eat? It says 'FOOD'" She was also very intrigued when she mistakenly typed "BLT, and the computer responded "YOU HAVEN'T TOLD ME HOW TO BLT." (Much later she was to say "let's teach it how to BLT," and "Gary, remember BLT?"

Laura, who is left handed, often had difficulty "driving the TURTLE," especially in distinguishing between left and right. (In her face project, the eye on the right of the drawing was called LEFTEYE which would have been accurate from the perspective of the face -- looking out from the display screen.) She often reversed letters in spelling (NOES for "nose", etc.)

At the same time, Laura demonstrated over and over again, that she did not like to make an analytical effort in her work. She could copy a "formula" successfully, and even have an idea why it worked, but she had difficulty in adapting it to a new situation, or changing it slightly. She rarely made a specific plan that she tried to carry through -- preferring to erase a procedure rather than edit it, and to clear the screen and start again with a drawing, rather than analyze what was wrong. She even developed a habit of typing POTS (PRINTOUT TITLES), every time she wanted to make any change in what she was doing. This had the effect of displaying a long string of procedure names on the screen, and totally "wiping out" her previous work. (Tina created the same effect by typing a string of carriage returns.)

It seemed important for Laura to "give the world the impression" that she "knew a lot," without making a real intellectual effort to learn. Along with this went the strategy of hiding what she didn't know, and when she did ask for help, only attending to the minimal amount necessary to solve the immediate problem -- i.e. get the procedure to do the right thing -- without concentrating on the underlying principles or debugging strategies.

2. Laura' Work in Turtle Geometry

During the first class, Laura had a good deal of difficulty "driving the TURTLE" -- choosing correct numbers for right, left and forward. She appeared "bored" with TURTLE driving activities, and the teacher introduced the idea of writing a random procedure, and repeating it -- an idea that Laura seemed to enjoy and understand -- but that she never tried for herself when working alone. When Laura got her first chance to work alone at a graphics terminal she worked without advance planning to build an elaborate open ended design. In the next class, she did another elaborate design involving many instructions, some of which had already disappeared from the screen before the design was complete. When she attempted to "capture" the design on paper so that she could teach it to the computer, she had already lost some critical steps from the first part of her exploration.

At the beginning of the next session, she was asked to plan a simple design by drawing it first on paper, then trying to draw it on the screen and finally teaching it to the computer. Laura decided to make a face, and drew the following sketch!

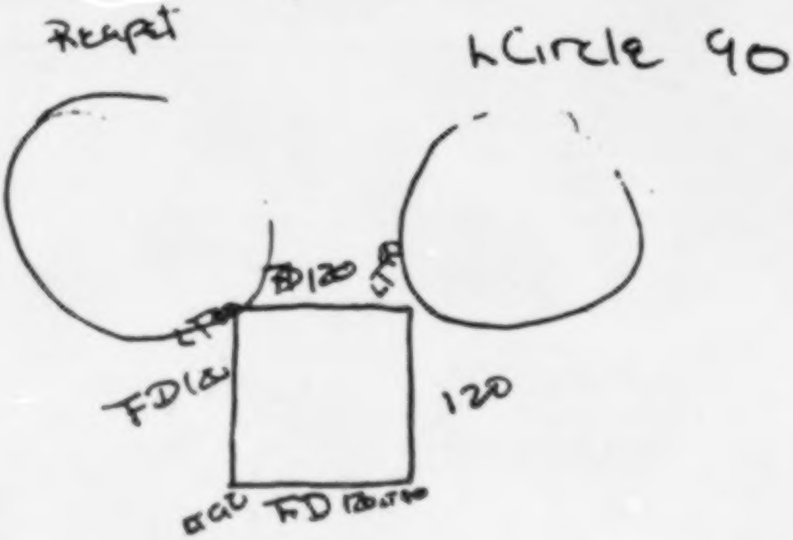


Figure 13.1

Carrying out this project was difficult for Laura. She had not yet written any procedures, and this project needed sub-procedures. In addition; all her previous designs had been "planned as they went along," with Laura working in "designer mode" -- try this -- then try that. Now she was trying out a fixed plan -- but she did not understand clearly enough how to manipulate the sizes of the elements -- circles and squares, and how to locate the turtle so that they would be in the correct positions. She needed a lot of help to work through this project.

```
TO FACE
1 NOES
2 RIGHTEYE
3 LEFTEYE
4 MOUTH
5 SQUARE1
END
```

```
TO RIGHTEYE
1 PENUP
2 FORWARD 60
3 LEFT 90
4 FORWWARD 40
5 RIGHT 90
6 PENDOWN
7 LCIRCLE 30
END
```

```
TO LEFTEYE
1 RIGHT 90
2 PENUP
3 FORWARD 80
4 RIGHT 90
5 PENDOWN
6 LCIRCLE 30
END
```

```
TO NOES
1 LEFT 90
2 FORWARD 100
3 RIGHT 90
4 SQUARE
5 RIGHT 90
6 FORWARD 20
7 LEFT 90
END
```

```
TO SQUARE1
1
2
3
4
5
6
7
8
10 PENUP
20 FORWARD 70
30 RIGHT 90
40 FORWARD 160
50 PENDOWN
60 LINE1
70 LINE1
80 LINE1
90 LINE1
END
```

```
TO MOUTH
1 PENUP
2 FORWARD 100
3 RIGHT 90
4 PENDOWN
5 FORWARD 90
6 HIDETURTLE
END
```

```
TO LINE1
1 RIGHT 90
2 FORWARD 225
END
```

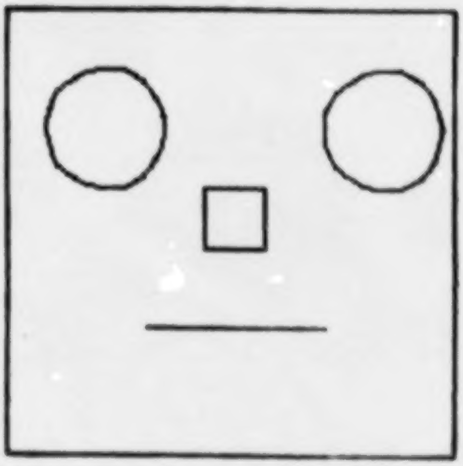


Figure 13.2

In retrospect, this project may have been an interuption of Laura's natural learning path. The things that she "learned" were not totally absorbed by her at this point. Laura was also put in the position of "needing help", which continued until the project was finished. On the other hand, she was very pleased with the result, and did have an opportunity to work much more carefully and critically with TURTLE manipulations. She learned to write procedures and sub-procedures, and was exposed to the idea of a superprocedure.

When Laura went back to "designing," she used circles of different sizes. Laura's next procedure, AROUND, a collection of different sized LCIRCLES all starting from the same point, was not constructed by following a specific pattern -- rather, Laura added circles in a somewhat random way: Large, smaller, smaller, smaller, smallest, largest, smaller, smaller, larger...as if she were studying the design and asking "what size circle would look good now?" Her procedure was copied directly from the screen. She numbered her steps by 10's for the first time.

```
TO AROUND
10 LCIRCLE 90
20 LCIRCLE 58
30 LCIRCLE 48
40 LCIRCLE 20
50 LCIRCLE 10
60 LCIRCLE 96
70 LCIRCLE 50
80 LCIRCLE 33
90 LCIRCLE 66
END
```

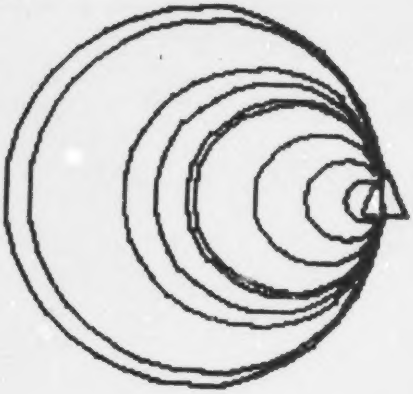
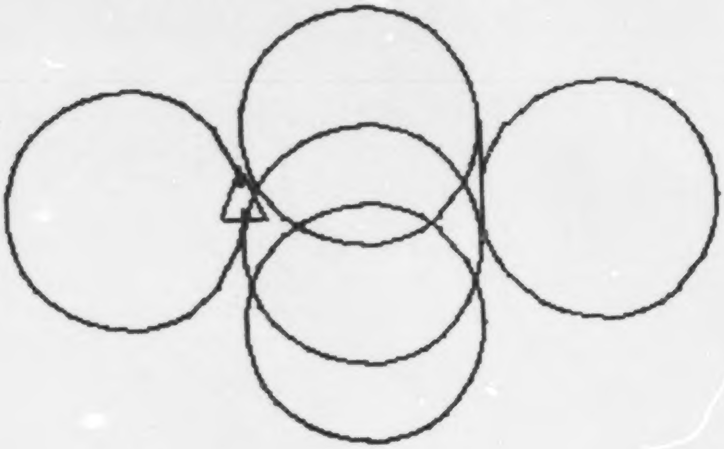


Figure 13.3

Next, she made a symmetrical procedure AROUND1 with RCIRCLEs, and then put them both together. Following this Laura went back to more free form experimentation in her "designer" mode -- try this...then see what looks good next.

Session eleven was one of Laura' best days. She developed a fairly complex design. She copied the steps in her notebook, and then taught the procedure to the computer. At the teacher's suggestion, Laura made one part of the design a sub-procedure, which simplified debugging. Laura had to do a lot of debugging, because she had difficulty copying correctly from her notebook. She worked a lot with the EDIT and PRINTOUT (PO) commands, comparing the steps as written in her notebook, with the steps as listed in the procedure. She did not debug by tracing through the procedure directly, to see what each step was doing.



GOE

Figure 13.4

Laura' working style seems to have been: Try something. If you like it, copy the steps down in your notebook. Then teach it to the computer copying the steps from your notebook. Try to be careful not to make a mistake copying. Although Laura realized that procedures could be changed if necessary, she did not see the procedure itself as the thing to be experimented with -- trying it out, and then changing it to make it do what she wanted.

During later classes Laura worked a lot with recursion, variables, stop rules, and a long language project -- a "mad lib" game. She did not return to "designer mode" until one of the last classes, when she picked up on one of her first project ideas -- making her initials.

It is clear from her dribble files that Laura worked on her initials by a process of trial and error -- then wrote down the correct steps in her notebook. She did not plan ahead or try to think about the easiest way to do it. As a result, she wound up having procedures with many more steps than necessary, as she retraced her course in finishing her letter. Her E, made of just four straight lines, had 15 steps. She reverted to incrementing line numbers by ones after reaching 100: 10, 20, ...100, 101, 102, ...107.) In addition, she had bugs resulting from mis-copying steps into and out of her notebook. The teacher suggested stepping through the procedure after printing it out and checking it against her notebook. Laura did not have the idea of stepping through the procedure on her own, (or didn't think she could do it without help, or didn't want to bother doing it without help.)

When Laura worked on her B (her last initial), she ran into exactly the same problems: little or no planning; poor copying to and from notebook; line numbers increase by ones above 100; confusion about whether to turn the TURTLE left or right, and especially no clear sense of how to debug by stepping through a procedure (playing computer), although this technique had been emphasized with her for five classes in a row.

Laura missed two of the last four classes (she was absent for one, and went on a class trip for another) and never completed her initials -- although she could probably have finished them, and created a superprocedure to draw all three, in about one more session.

3. Laura's Work With Recursion and Variables

During classes 12-16, Laura was introduced to recursion, procedures with inputs, recursion with fixed inputs, and recursion with varying inputs. In later classes she returned to do some more work with variables using recursive procedures with two inputs.

In session 12 both recursion, and the use of variables were introduced to Laura. She was shown a procedure:

```
TO TWIST
10 LCIRCLE 40
20 RIGHT 30
30 TWIST
END
```

She then wanted to make TWISTs of different sizes and was shown how to make the size of the circle a variable. She wrote:

```
TO TWIST2: SIZE
10 LCIRCLE :SIZE
20 RIGHT 90
30 TWIST2:SIZE
END
```

She seemed to understand what she was doing. Although she had introduced the RIGHT 90 in line 20, she seemed surprised that the shape of her circle design was different from the preceeding one. She understood that she could vary the size of the design by varying the input to TWIST2.

In the next session Laura continued to write procedures with a variable SIZE. She began to indicate more confusion. She usually left the :SIZE out of the procedure title. She tried to use EDT (EDIT TITLE), but used it incorrectly and disregarded error messages. She wrote a procedure called TO SQUARE: SIZE, which did not use a variable :SIZE within the procedure. And, when copying a procedure, TO SQ:SIZE, she made all the forward steps FORWARD 66:SIZE, not understanding that :SIZE replaced the specific forward step. Once she had the SQ :SIZE procedure defined correctly, she was able to make a very exciting design using SQ 1, SQ 2, ...SQ 82.

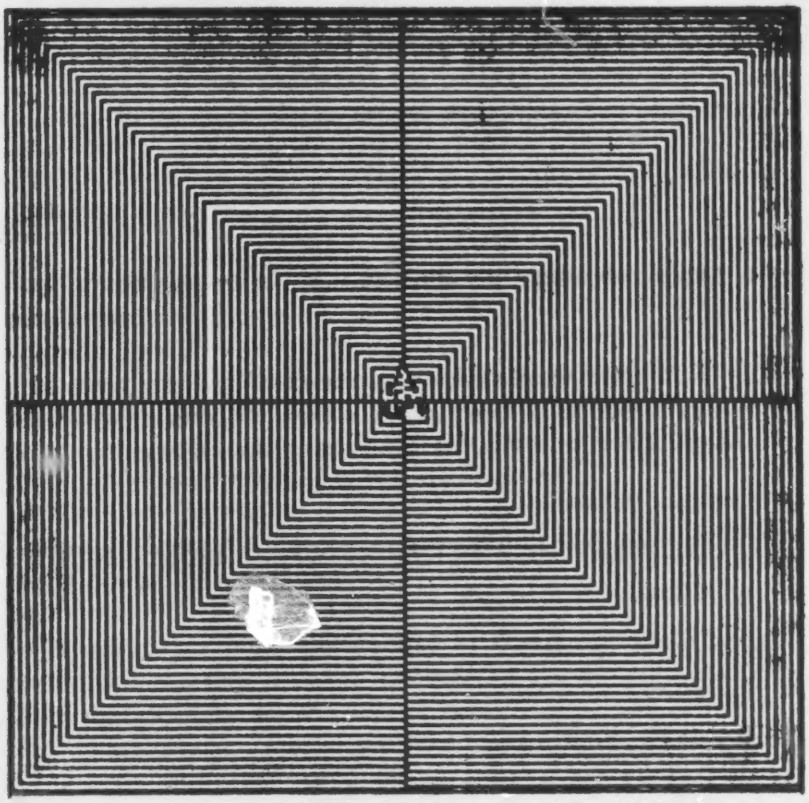


Figure 13.5

To capitalize on this discovery, the following procedure was shown to Laura at the next class (14):

```
TO GROWSQ :SIZE
10 SQ :SIZE
20 GROWSQ :SIZE+1
END
```

However, when Laura tried to copy this, she typed:

```
TO GROW SQ :SIZE
10 SQ :SIZE
20 GROW SQ :SIZE
END
```

She forgot to type in the +1, despite a very careful explanation by the teacher of how the procedure worked. Also, since she left a space between GROW and SQ:SIZE, she kept getting error messages when she tried to use the procedure. She did read the error messages, and tried several ways of typing the procedure, to try to eliminate the error. She tried GROW SQ, GROW SQ 1, GROWSQ 1, GROW SQ 1 100, GROW 1 100, none of which worked. Finally she asked for help, and the teacher suggested she erase the procedure GROW and copy it over, making GROWSQ one word. She had been resourceful in trying different ways of typing the procedure names, but she did not look at the procedure itself to see what was wrong. Another example of Laura' basic working style; lots of trial and error -- no analysis!

Later in the same period, Laura was trying TWIST 80, TWIST 40, TWIST 300. All of these produced the same figure, since TWIST was a fixed instruction procedure. When Laura finally asked for help, she was asked to print out the procedure and look at it to see why it always made the same shape. Once again, Laura had tried different things, but had not looked at the procedure to see what was wrong. Still later in the same class, she was making a procedure to draw a letter T for a friend, whose name was Tina. She made two attempts TINA, and TINA1, neither of which worked. In neither case did she look at the procedure and try to analyze it.

The next session was a group lesson about debugging, the use of PO (PRINTOUT) and step by step analysis of a procedure. Laura spent the period playing around with old procedures, trying to copy other kids procedures from the bulletin board (without copying the subprocedures needed), and in general wasting time. This was interpreted by the teacher as "boredom" and he suggested a new project. Laura agreed to try something new, but in her notebook that day she wrote "By the way, I am not board"(sic). She may have been confused about what had happened when she tried to use variables -- and what she needed was probably more simple projects using variables in procedures and subprocedures. Laura had been given more ideas than she could absorb, and this was the message she was giving -- not boredom.

Instead of giving Laura an opportunity to explore things she already knew, and to consolidate what she was learning about variables, she was launched into a new project: writing a program to produce "Madlibs". This was a good learning experience for her. She had to clarify her ideas about nouns, verbs, adjectives and adverbs and how they are used in English -- as well as to make up a simple story, and choose lists of words that would make the story funny. On the other hand, it presented her with a whole new set of things to be confused about, and required a lot of precise typing. Although Laura was able to understand the language aspects of the procedure, it is unclear what she understood about the programming required, which involved the use of OUTPUT, MAKE and a bit of list processing. Laura was pleased when the project was completed -- but she hadn't been able to do much of the programming herself or understand how the procedures worked.

4. Conclusions

If Laura was to continue as a LOGO student, we would reconsider our teaching strategy as follows:

> --encourage her to continue explorations with TURTLE commands. She still had a lot to resolve in the areas of left/right discrimination, analyzing sequences of steps to see their effect, and attention to detail in copying.
>
> --stress planning of simple projects like initials, encouraging more use of subprocedures.
>
> --go back to simple applications of variables, again with stress on planning -- what is supposed to change? where does it belong in the procedure? what name is chosen for the variable? etc. Then there would be examples of the use of procedures with variables as subprocedures, in both recursive and non-recursive situations. Laura was close to understanding these points -- but got presented with too much, too soon.
>
> --One area which Laura did not really get into, which might excxite her as a designer, is repetition of a random set of commands.
>
> --It might be useful to experiment with the use of an automatic drawing procedure like DRAW, so that Laura could

experience success with some of her more elaborate designs.

Given Laura' avoidance of "cognitive risk"; her reluctance to reveal her confusions; and her desire to appear to "know" everything instantly, she would always be a difficult child to teach. On the other hand, situations with more stress on ways she could plan and predict outcomes, and fewer sources of confusion introduced from outside, could probably help Laura assimilate some of the problem solving skills which she avoided.

14. Monica

Monica is considered to be an "average" student by her teachers. Her most recent school administered national achievement test ranking was in the 56th percentile. Her teachers find that she prefers to base her activities solidly on things she knows, rather than to strike out into new areas.

1. Monica's Working Style

Monca's work in the LOGO class followed this pattern as well. She learned the basics of LOGO quickly and started off with a burst of enthusiasm and confidence. She began with a very good sense of TURTLE state -- she seemed to know where a figure would be drawn next, had a good sense of how far to turn, etc. She was fascinated by making a figure, rotating it, and repeating the process over and over. She easily adopted the use of simple recursion to do this, and created many projects of this type.

As the classes went on, Monica limited most of her work to this particular mode, and had a hard time getting beyond it. Almost all her work was in the style of "figure, turn, figure, turn.." She learned to use variables, to make the figure turn different amounts, and had some exposure to stop rules, which she didn't quite master. Monica did not get involved in long term projects of any kind, or show much initiative in breaking out of the "mold" in which she had placed herself.

Monica had a very close relationship with Kathy, the other girl in her class. (According to their classroom teacher, they did not have a close relationship outside of the LOGO class.) The two girls often consulted together, borrowed ideas, worked on the same, or similar projects, and asked for and offered help to each other throughout the classes. Their relationship was normally quite "mutual" with a lot of give and take, although leadership shifted back and forth. Different observers who had come for a single visit observed: "teacher-student relation between Monica and Kathy. Monica tells Kathy what to do and Kathy always goes to Monica to make sure she has done it right..." or, another time: "Monica was at a loss as to what to do with herself... at last Kathy arrived. Kathy found a worksheet for her...(she) got upset, panicked and ran to Kathy for a new thing to do." Another observer: "K and M work together very constructively -- each on top of things, making suggestions". This last observation corresponds most closely to our sense of their overall relationship throughout.

As the classes went on, Monica tended to have fewer ideas of what she wanted to do. She would borrow ideas from Kathy, from the bulletin board, or from a booklet of projects, often copying carelessly, by rote, not thinking about what the steps were supposed to do. Monica did not have much of an inclination to plan, to think ahead, or to debug her work. If something didn't work out the way she wanted it to, she would often just forget about it, leaving a bunch of useless procedures in her file, along with the good ones. She rarely asked for help. Although she had been shown how to use ED (EDIT) to change procedures at an early point, she rarely chose to use it. Her procedures tended to be short and simple. If they didn't do what she wanted, she'd forget them. During the last few classes, Monica expressed an interest in debugging a rather lengthy procedure, her HAT procedure, that drew a Christmas tree, so that the stump would be "straight". She wanted to change it from:

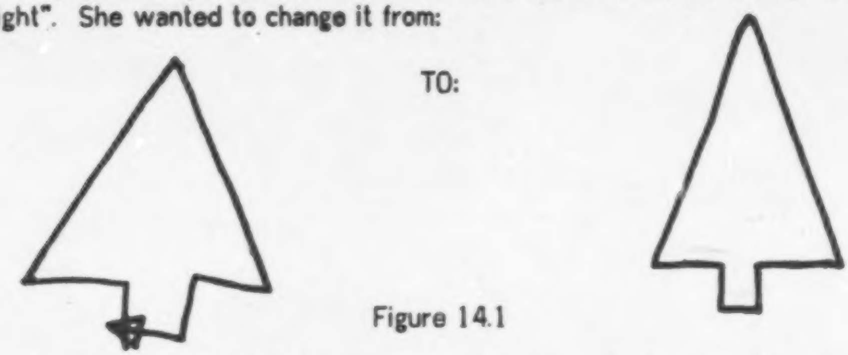


Figure 14.1

This was her first real interest in any procedures longer than a few lines that did not simply repeat a few fixed steps.

Monica's use of names was erratic. Her "HAT" made a "Christmastree." BOX, TRI, BUS, BUSWHEEL, BUSWHEEL2, HOUSE, HOUSE4, related to specific objects; HORSE, WOW, WISHWOW, BOODLE, HOTHOUSE, were fairly random. She seemed to have difficulty choosing names, as she had difficulty chosing projects.

Monica kept thorough notes of her work by writing down every procedure in her notebook, either before or after trying it out.

2. Monica's Work in Turtle Geometry

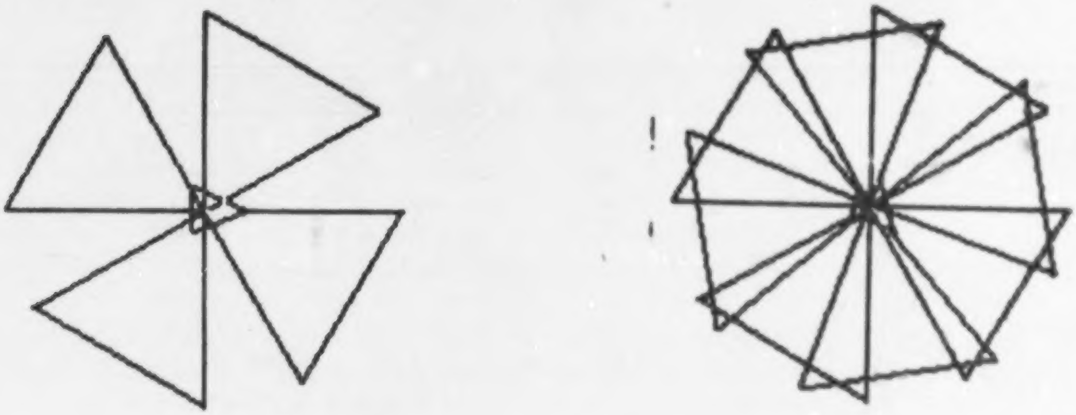
During the first few classes, when the group of four children worked together, Monica demonstrated a good understanding of TURTLE state. By considering where the TURTLE was, she was able to predict where the next procedure would occur. This was especially useful, because the projects that the children were doing involved making designs with squares, using a BOX procedure. In the sixth class, Monica and Kathy worked together, putting a BOX and a TRIANGLE together to make a HOUSE. Monica had a very strong sense (much better than Kathy's) of how much to turn the TURTLE, to get the two figures to line up.

On the other hand, when Monica tried to make the TURTLE draw a triangle, she had great difficulty separating the different variables. She worked steadily for an hour, trying to make a triangle that would close. Her problem was, that she worked without an effective system. She had to deal with five different variables (three lengths and two angles). She had a hard time fixing on which one to vary, and so, kept getting close to a solution, only to have her next attempt produce something quite different. She used two different strategies as she worked, and kept switching between them. She got quite confused about what was happening, and never succeeded in getting the triangle to close. Still, she stuck to the task doggedly and came remarkably close to a solution without actually getting one.

In the seventh class, Monica copied a state transparent equilateral triangle procedure, and began to experiment with the effects of putting rotations between triangles. She put a whole series of triangles rotated at different angles, on top of each other.

It was suggested that she give names to some of her designs such as TRI90 for a series of repeats of TRI, LEFT 90; or TRI40 for a series of repeats of TRI, LEFT 40 (both sequences she had used in the previous class). Monica understood the idea of making each design a separate procedure, but she had her own approach to naming them:

```
TO TRI4            TO TRI42           TO TRI442
1 TRI              1 LEFT 40          1 TRI4
2 LEFT 90          2 TRI4             2 TRI42
3 TRI              END                END
4 LEFT 90
5 TRI
6 LEFT 90
7 TRI
END
```

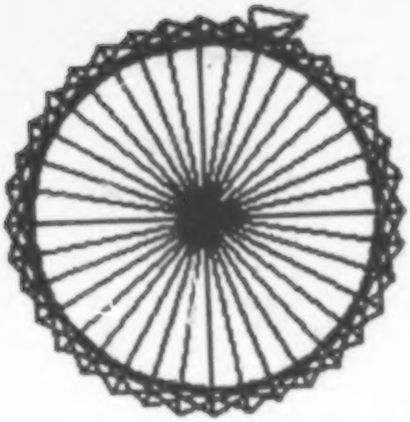


TRI4 TRI442

Figure 14.2

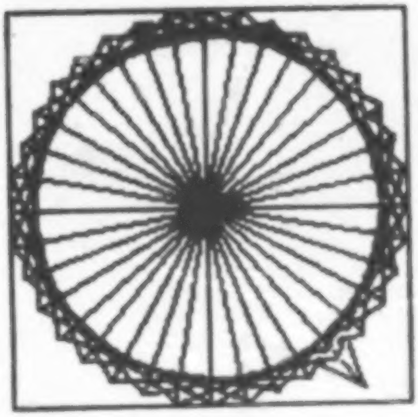
The rest of the period, was devoted to a lengthy series of repeats of TRI, LEFT 10. Monica's plan was to complete a circle of these with this shape, and then to teach it to the computer as a procedure. After 13 repeats, a half circle was completed, and Monica concluded that 26 repeats would produce a full circle. At this point, Monica was shown how to use recursion as an "easier" way to accomplish what she wanted to do:

```
TO FAN        and        TO FANBOX
1 TRI                    1 4BOX
2 LEFT 10                2 FAN
3 FAN                    END
END
```



FAN

```
TO FANBOX
1 4BOX
2 FAN
END
```



FANBOX

Figure 14.3

3. Difficulties with Debugging

On another occasion Monica mis-copied one of Gary's procedures from the bulletin board. The title should have been FOO2 instead of FOO. Her FOO had no graphic effect, but produced a "NO STORAGE LEFT.." error message.

```
TO FOO
10 FOO
20 FOO
30 FOO
40 FOO
50 FOO
60 FOO
70 FOO
80 FOO
90 FOO
100 FOO
110
120 FOO
130 FOO
END
```

This was the first time that Monica had not tried each step of a procedure, before teaching it to the computer. This time she had copied a procedure line by line, without realizing that she also needed a subprocedure, FOO. She did not ask for help, or try to debug FOO in any way, but ignored it and went back to a previous exploration.

In subsequent classes Monica continued to work in ways which were becoming a definite pattern: she did more work along the line of shape, rotation, shape, rotation, etc. She also avoided debugging procedures which did not do what she wanted. During a later class she copied some more procedures from the bulletin board, and from a project book. Most of these procedures did not work, either because of errors in copying (like the mistake she had made with FOO) or because she did not pay attention to the subprocedures needed in each case.

Monica recorded some of her difficulties in her notebook, without any attempt to analyze them: "Today I made a DOODLE... and I tried two DOODLES but it wouldn't work out too well..and I tried to make a slinky."

```
TO DOODLE                          TO SLINKY
10 DOODLE                          10 CIRCLE
20 DOODLE                          20 FORWARD 10
30 DOODLE                          30 SLINKY
40 DOODLE                          END
END
```

In going over the dribble files for these classes, it became evident that Monica was not looking carefully at what she was doing; that she was not editing or debugging; that she was following certain patterns blindly without thinking about them; and that she had run out of ideas for projects. For the next class, the teacher prepared a collection of buggy procedures, discussed each procedure with Monica, and then had her try out the procedures to see what the computer would do in each case. The purposes were to get Monica to notice and focus on the messages sent by the computer in buggy situations, and to help her understand some of the particular kinds of bugs that she was experiencing.

Following this lesson, Monica began doing some debugging, but still had difficulty understanding how to use the EDIT command. Her teachers worked through one problem with her using PO (PRINTOUT) and ED (EDIT), and analyzing a procedure step by step. In this way, Monica was given a model of how she could work through other situations without help.

In class 13, Kathy and Monica were introduced to the idea of variables, by giving them a variable square procedure, SQ:SIZE together with a sheet of possible project ideas. Monica and Kathy both chose to make a bus (Figure 14.4), starting with two squares:

```
TO BUS
1 SQ 40
2 LEFT 90
3 SQ 80
END
```

Monica worked on the bus for parts of two periods. Once again, she had problems editing, editing the line, rather than the procedure. For example, her BUSWHEEL procedure was

```
TO BUSWHEEL
1 BUS
2 LEFT 90
3 RCIRCLE
END
```

When she typed BUSWHEEL, she got the error message; "RCIRCLE NEEDS MORE MORE INPUTS AT LEVEL 1 LINE 3 IN BUSWHEEL." Her response was to type ED RCIRCLE. Clearly she was reading and interpreting the error message, and using the information to try to debug her work, but she did not understand the proper use of EDIT.

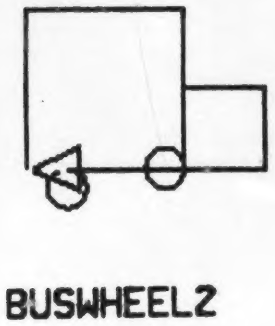
Another bug was more surprising. She had a problem with the TURTLE state, in aligning the wheels properly. Her BUSWHEEL procedure had become:

```
TO BUSWHEEL
1 BUS
2 LEFT 90
3 RCIRCLE 10
END
```

and when used to make a complete bus:

```
TO BUSWHEEL
1 BUSWHEEL
2 RIGHT 90
3 FORWARD 60
4 RCIRCLE 10
END
```

makes a bus with a wheel out of position, shown in Figure 14.4.



```
TO BUSWHEEL2
1 BUSWHEEL
2 RIGHT 90
3 FORWARD 60
4 LCIRCLE 10
END

TO BUSWHEEL
1 BUS
2 LEFT 90
3 RCIRCLE 10
END

TO BUS
1 SQ 40
2 LEFT 90
3 SQ 20
END
```

Figure 14.4

Either Monica did not notice that the wheels were at different levels, didn't feel it was a problem or didn't want to bother with it. She ignored it, deciding that the project was complete. She went on to other work, going back to her old pattern of procedure, rotation, procedure, to produce some more nice designs (see, for example, Figure 14.5 for a way that Monica used her BUS procedure in a more familiar mode.)

```
TO BUS
1 SQ 40
2 LEFT 90
3 SQ 50
END
```

```
TO 4BUS
1 BUS
2 BUS
3 BUS
4 BUS
END
```

```
TO STAR
1 4BUS
2 RIGHT 40
3 4BUS
END
```

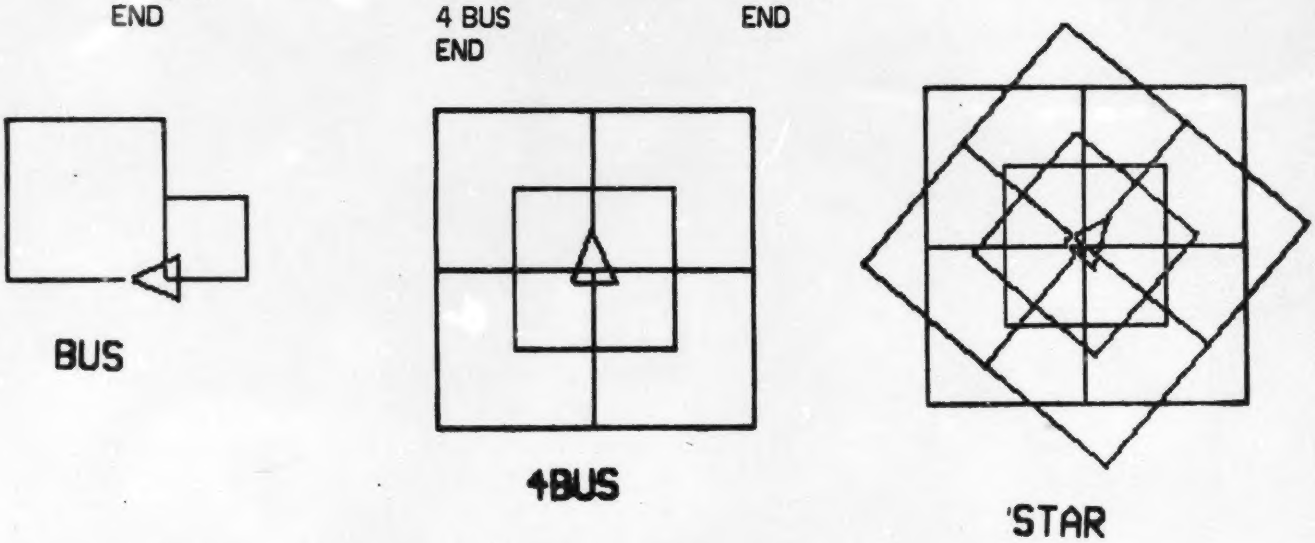


Figure 14.5

In the next few classes Monica continued to use the rotation idea, sometimes with recursion, to make procedures like:

```
TO DESIGN
1 STAR
2 RIGHT 40
3 DESIGN
END
```

```
TO DOG
1 BOX
2 RIGHT 70
3 DOG
END
```

Her projects were short, and she did not have to do any debugging. She did borrow one long procedure -- Kathy's XMASTREE, which Monica decided to call HAT. In copying this procedure she needed to do some debugging and asked for help with the EDIT command.

4. Understanding the Effects of Angles

Although Monica had been using rotations to produce designs, she had not discovered any consistent effects of using particular angles. She had used rotations of 10, 20, 40, and occasionally 70 or 90 degrees in her designs.

Although she seemed to prefer dense designs, she seemed to have no way of predicting the effect of using a particular angle, or the sense that certain special angles might produce nice designs that closed in a predictable way.

To help her explore the effect of different angles Monica was shown some new procedures that used a variable angle:

```
TO SPINBUS : ANGLE
10 BUS
20 RIGHT :ANGLE
30 IF HEADING = 0 STOP
40 SPINBUS :ANGLE
END
```

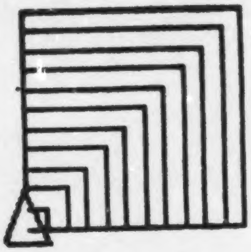
The use of the stop rule, was another suggestion to help Monica focus on when the design was "complete." Monica used this model to define procedures that would rotate her HAT procedure, her WOW procedure (a series of nested squares) and her HOUSE procedure.

In typing these procedures, Monica had some difficulties with syntax. She usually left out the :ANGLE in the procedures title, and was shown to use EDT (EDIT TITLE). Now she was using ED properly, but had difficulty with EDT. She did, however, correctly interpret the error messages, and debug the procedures on her own.

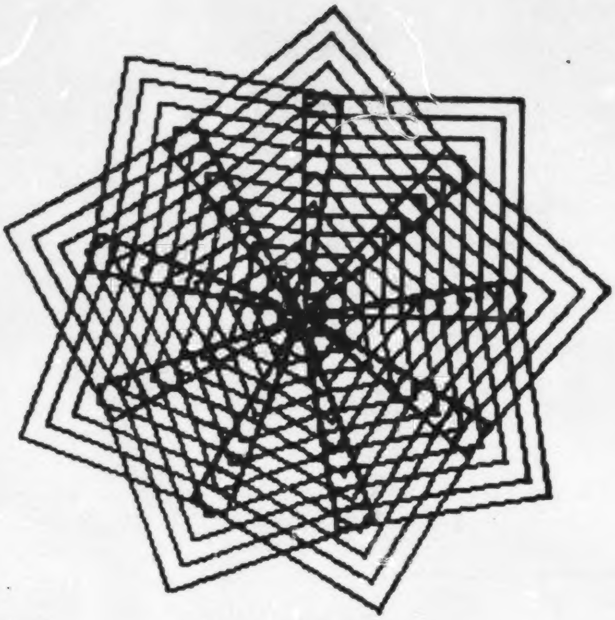
In the next class, Monica was asked to choose one of her procedures, and experiment with varying the input, keeping records of the results. She chose to use the procedure

```
WISHWOW :ANGLE
```

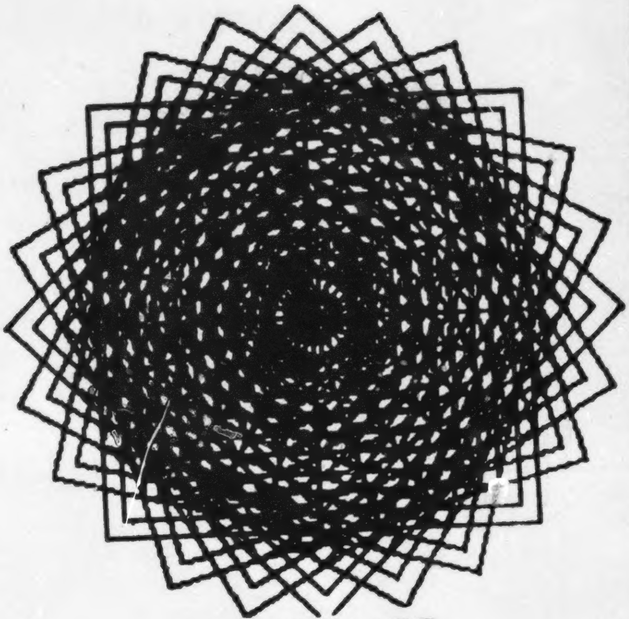
which rotates a group of nested squares.



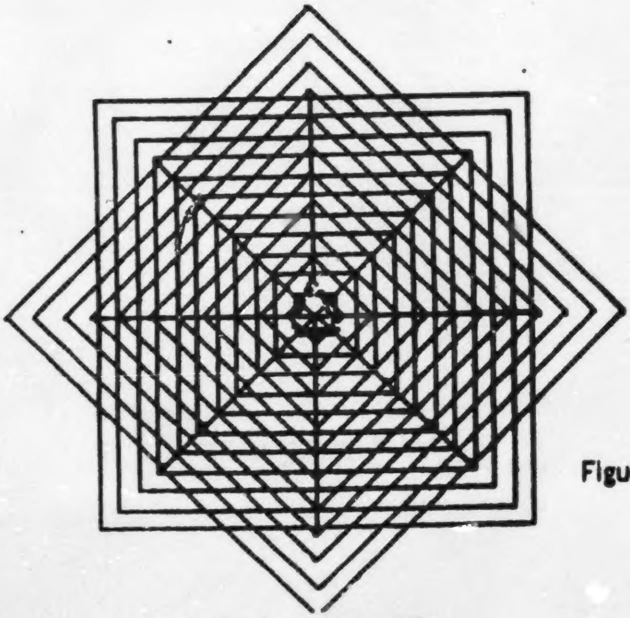
WOW



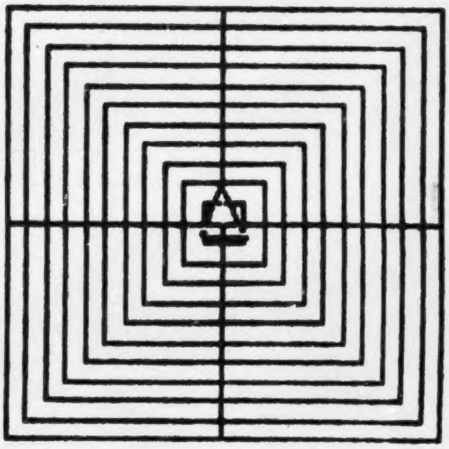
WISHWOW 160



WISHWOW 165



WISHWOW 45



WISHWOW 90

Figure 14.6

She kept meticulous notes in her notebook, describing what the shapes looked like, how they grew, and comparing them with similar shapes. For example:

> "WISHWOW 160 looked the same as WISHWOW 40. It had thin cones and there were 9 of them.
>
> WISHWOW 165 had thin webbed cones and you couldn't really see them that good.
>
> WISHWOW 190 had cones but they looked like they didn't close up. And it was fatter than other ones. It had more squares and cones. The cones were thin. And close together.
>
> WISHWOW 45, WISHWOW 90. These 2 look almost the same but WISHWOW 45 looks like it goes twice around instead of once. And the cone shaped things on the sides are bigger than the WISHWOW 90 ones."

Clearly, Monica was making some interesting "discoveries" about angles. If she had gone on, she could have compared two procedures -- for example, WISHWOW and SPINBUS, with the same inputs. She could also have used other inputs which involved simpler rotations (such as 45 and 90). A chart could have been helpful to organize the information she was gathering. All these things would have allowed her to consolidate her discoveries about angles.

5. Conclusions

Because each of Monica's individual "projects" had been very short and simple, Monica had not developed a sense of urgency about "finishing" any of her work before the end of the series of classes. Although some important continuations, consolidations and clarifications were beginning to occur for her. Monica, herself, had no such sense of continuity. Although she probably would have enjoyed continuing, she was also quite content to stop her work at this point.

Throughout the classes, Monica seemed to be most comfortable learning by direct imitation of examples or models supplied. In this way she learned to write procedures utilizing simple recursion, variables and stop rules. If the context was shifted, or a small mistake led to a bug, Monica was often stuck. She usually chose not to analyze her mistakes, nor did she undertake long projects requiring advanced planning, or a large number of subprocedures. Near the end of the series of classes Monica gradually began to be comfortable with editing, and to understand how to analyze a procedure in a step by step fashion.

Monica's investment in any particular project was slight -- she could easily discard it without debugging if a problem occurred, and go on to a new activity, which might prove succesful. She had a large number of small procedures, which allowed her to feel successful most of the time she was working, without having to confront her confusions. Her dribble files show numerous ideas sidetracked without debugging -- and apparently without any strong feelings of disappointment. In this way, Monica was able to function comfortably in an environment which was more complex than her understanding of it. When she did successfully assimilate a concept (as she was beginning to do with editing and debugging) she did not look back to old problems, to see if she could solve them now with her new tool. Rather, she unselfconsciously applied the new idea to whatever new problems arose. The old problems had been conveniently forgotten.

It is possible that Monica would have benefitted from being able to use a carefully designed set of worksheets structured to lead her from one concept to another with many small projects along the way.

15. Ray

Ray is a student who has been diagnosed by school personnel as having "learning disabilities". He is tutored individually by a learning disabilities specialist several times each week. His teachers feel that at the beginning of the year he was noticeably "slipping" in his seriousness as a student. (His 6th grade achievement test placed him in the 24th percentile, nationally.)

Although Ray was initially quite successful in controlling the motion of the TURTLE, he held himself somewhat aloof from the activities in the LOGO classes. As a result, he never succeeded in writing a procedure without assistance, although he had considerable success (with help) on several projects such as drawing and animating a rocket (sessions 13-15) and in using the computer with procedures that enabled him to explore geometric shapes. In general, Ray had success using the computer in two kinds of situations: when a teacher was helping him intensely during a session, and when he was working in a way that required him to remember only one variable at a time.

The teaching strategy for Ray was to try to structure situations in which he could be successful without a lot of help from the teacher, since Ray would usually "forget" what to do when the teacher was no longer present. For the longest time, Ray did not engage in much "free experimentation" with the TURTLE. But towards the end of the series of classes (session 19) he was given a POLY procedure which requires two inputs to produce a series of closed geometric shapes, and a SPI procedure which required three inputs and produced a variety of spiral shapes. Ray gradually learned how to control the inputs to produce certain shapes in a predictable way. For the first time, he began to experiment in a purposeful way, to write things down in his notebook, to use those notes to remember successful designs. He began to gain confidence in his ability to control the computer. He invited a friend to class -- together they had a very exciting time exploring the shapes produced by the POLY and SPI procedures. Ray's teachers also reported a noticeable improvement in his attitude in class, which they attributed partly to his feeling of success in the LOGO classroom.

1. Ray's Working Style

From the first day he came to class, in session 2, Ray insisted on remaining "aloof"; preventing himself from feeling personally involved with the LOGO activities. Although he started off quite successfully, and actually was generally successful under close supervision in his first programming project -- causing the computer to draw his initials -- he maintained his "cool" until the last four or five sessions, when he began to become interested in what he was doing.

Before summarizing his work, it would be useful to list some of his techniques for maintaining and reinforcing his posture of aloofness from the activities. He began by coming four or five minutes late for each class. By coming to class late, he could be sure that the teacher would be already working with another student. Then he could wait several more minutes for the teacher to remind him how to LOGIN, and offer a suggestion for his day's work. He often walked into class whistling loudly, blatantly disregarding anything else that was going on. He made a point of always leaving a few minutes early, and as he worked on activities, he would look at the clock, to see if it was time to leave yet.

Ray also made a point of not remembering how to do things. He would not write things down in his notebook, and when asked to consult a reference sheet or an entry in his notebook, he would usually just sit, and wait for help. This was his way of reinforcing a sense of helplessness, of "I can't do it", of dependency on the teacher. Rather than maximizing his use of available resources, Ray deliberately minimized them.

Ray refused to learn the details of operation of the system and the language. Not until class 12 did he LOGIN by himself. He never wrote a file without assistance. Likewise, he never wrote a procedure without help. Although he was introduced to the REPEAT command, and used it to make TURTLE designs that were quite pleasing to him, he never remembered the format for using it, and would not look it up.

We are using words like "refused to learn", "techniques for maintaining..aloofness," "made a point of not remembering," because we are convinced that these were definite strategies Ray used, to protect himself from involvement, rather than an inability to concentrate or learn.

At the same time, Ray was clearly intelligent, attractive and charming. He had a special aptitude for music -- he enjoyed spending time tapping rhythmically, whistling, and improvising on the piano.

Ray appeared to have a strong fear of failure, and especially of appearing to fail. His strategy for coping with this has been the "class clown" approach -- act a little bit silly, charm everyone, and above all, don't let anyone know you're trying. Our teaching strategy with Ray was to try to structure situations so that he would be successful, and develop a sense of confidence -- a sense that he could do it. This was done both by setting up a special animation project for him, (which in the end required too much teacher help) and by trying to set up situations in which he could be sucessful with very little input. The latter approach proved to be the best for Ray, as will be seen from the detailed description of his work.

2. Ray's Work in Turtle Geometry

Despite his difficulties, Ray was interested in the computer, and its power. He showed a good deal of "natural ability" in Turtle Geometry. Ray was very successful in directing the TURTLE, estimating quickly and accurately, both angles and distances. His first project was making his initials, and he carried it out quite successfully, combining skill in Turtle Geometry with a quick understanding of using the keyboard, and an understanding of how to write procedures. (See figure 15.1)

```
TO RG
1 PENUP
2 LEFT 90
3 FORWARD 70
4 RIGHT 90
5 PENDOWN
6 R
7 PENUP
8 LEFT 100
9 FORWARD 100
10 RIGHT 60
11 PENDOWN
12 FORWARD 70
13 BACK 70
14 RIGHT 90
15 FORWARD 90
16 LEFT 90
17 FORWARD 70
18 LEFT 90
19 FORWARD 40
20 LEFT 90
21 FORWARD 30
END
```

```
TO R
1 FORWARD 100
2 RIGHT 90
3 FORWARD 50
4 RIGHT 90
5 FORWARD 40
6 RIGHT 90
7 FORWARD 50
8 LEFT 140
9 FORWARD 87
END
```

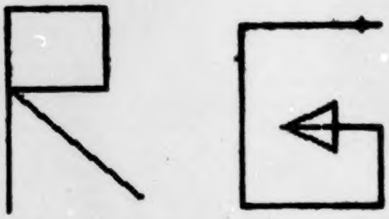


Figure 15.1

As Ray went on beyond this project, however, there appeared to be many difficulties impeding his progress. He would rarely work purposefully unless the teacher was present. All his further Turtle Geometry explorations appeared to be random. He did not remember how to write a procedure, and refused to "look it up", in his notebook. He was shown how to make a procedure out of two or three TURTLE steps, and the use of REPEAT to make designs. He made several simple designs -- SAM, TIM and JOE. Still, he did not remember from time to time, how to make a procedure, or how to use the REPEAT command. Again, he refused to look it up in his notebook.

After completing his initials project Ray's work for the next seven classes was characterized by short bursts of activity -- especially when the teacher was present -- and frequent visits to the water fountain, and to the piano in the next room. Ray was absent for four of the first eleven classes. What Ray <u>did</u> do successfully on his own was to experiment with different inputs to REPEAT. He would use sequences like REPEAT [SAM] 90, REPEAT [TIM] 30, REPEAT [JOE] 20, REPEAT [SAM] 30, etc. (See Figure 15.2)

```
TO SAM                      TO TIM              TO JOE
1 FORWARD 17                1 FORWARD 19        1 SAM
2 RIGHT 90                  2 RIGHT 90          2 LEFT 150
3 FORWARD 29                3 FORWARD 36        3 TIM
4 LEFT 56                   4 LEFT 61           END
END                         END
```

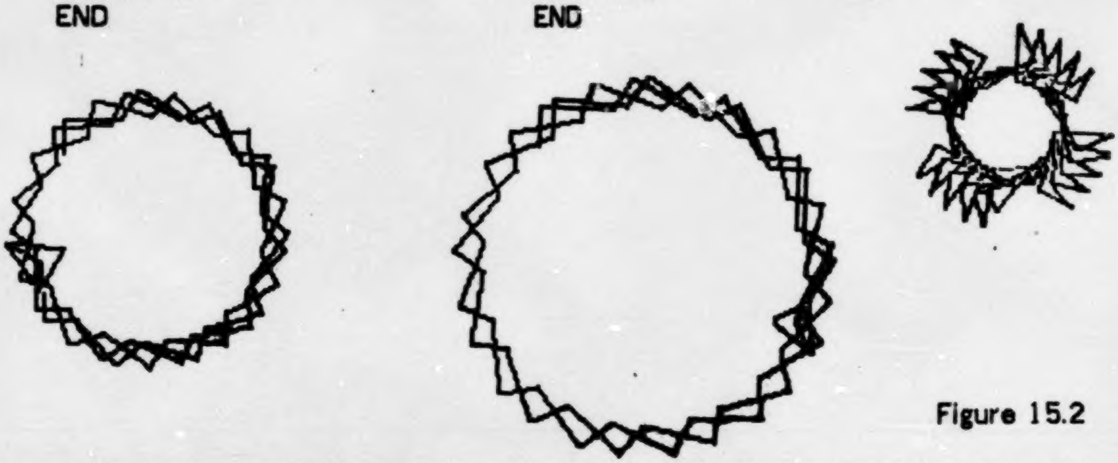


Figure 15.2

After doing some of these, he'd return to ones he liked. It seemed that he could focus for a short time on the task of choosing inputs to REPEAT, although he could not remember how to use REPEAT from class to class. After a few classes, however, this approach had lost interest for Ray, and it became clear that it had not resulted in his being able to write procedures independently, as had been hoped.

Next, the teacher suggested an animation project. Ray agreed, and decided to animate a rocket. One session was spent drawing the rocket. The teacher spent a major part of the period working with him, first helping him figure out how to draw a triangle (see Figure 15.3), and then helping him organize the task of teaching his rocket to the computer. His ROCKIT procedure had two subprocedures, TOP (triangle) and ROCK (rectangle). (See Figure 15.3)

```
TO ROCKIT          TO TOP             TO ROCK
1 TOP              1 RIGHT 90         1 LEFT 30
2 ROCK             2 FORWARD 60       2 FORWRAD 90
END                3 LEFT 120         3 LEFT 90
                   4 FORWARD 60       4 FORWARD 60
                   5 LEFT 120         5 LEFT 90
                   6 FORWARD 60       6 FORWARD 90
                   END                END
```

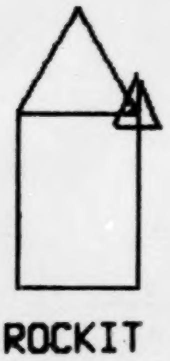


Figure 15.3

In the next class Ray and his teacher worked through the process of animating the rocket. He decided that his ROCKIT was too big, so he was helped to make a smaller, scaled down version, which he called FB. His procedure to move the rocket was called NKP. Ray was introduced to the idea of SNAPs. He understood how they were used to animate the motion -- but (of course) he had difficulty remembering the format for using them.

In the next class the teacher again spent a great deal of time with Ray. He was not especially interested in varying the speed of the animation but he was interested in making the rocket turn. Ray and the teacher printed out his animation procedure NKP, and figured out together where he could put a turn command, to make the rocket turn on the screen. Ray experimented on his own with different inputs for the turn: RIGHT 300, RIGHT 66, RIGHT 2, RIGHT 3, finally settled on RT 9, as the largest number he could use without having the rocket go off the screen.

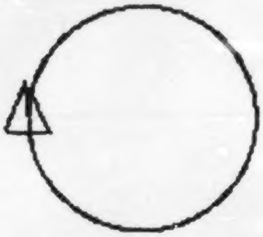
In analyzing the dribble file from this class we began to be strongly aware that Ray had been consistently successful in activities that required varying only one parameter at a time. We were now able to give serious consideration of how to get Ray involved in a more consistent relationship with his work.

3. Explorations with POLY and SPI Procedures

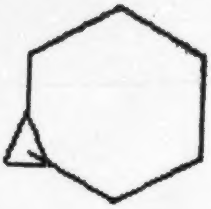
We finally decided to give Ray a POLY procedure, with a simple stop rule, and let him experiment with changing the inputs. In this way, he would have only one thing to consider -- the choice of numbers to make interesting designs. This was a successful choice and he continued to work with POLYs and a POLYSPI type procedure for the next six classes.

In working with POLY, Ray began by choosing inputs based on "number patterns" rather than on the effects produced by the POLY procedure itself. For example, during one class, the POLY inputs he chose were:

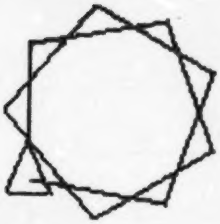
SIZE	ANGLE	
556	889	increasing and decreasing sequences
765	987	
567	987	
1000	2000	
999	999	
1	1	he was intrigued by the fact that
2	2	these made circles
3	3	
50	60	surprise! hexagon!
70	80	
70	89	he really likes this
70	90	he was shown this one, to go with it.



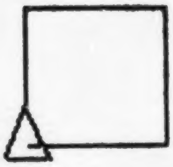
POLY 1 1



POLY 50 60



POLY 70 80



POLY 70 90

Figure 15.4

In next class, the teacher made a point of showing him that the first number effects the size, and the second number the shape. His work was still based on number patterns.

SIZE	ANGLE	
678	987	
70	89	an old friend
40	40	
50	50	same numbers
60	60	
78	93	
1	1	
1	34	makes a bright dot
98	89	reversing digits
567	123	
1000	1000	same input
200	7820	
678	876	reversal
9000	9000	same
765	897	
850	850	same
100	850	here he was shown the effect of
200	850	changing the first input, while keeping the second constant.

In the next class Ray was shown a spiral procedure. For the first time, Ray began to experiment with the effect of the changed input, by varying the rate of growth of the spiral. First Ray experimented with small numbers like SPI 1 1 and SPI 3 4. Even with inputs like SPI 22 33, Ray found the shapes "boring", because they all went off the screen so fast. The teacher suggested larger numbers for the second input, and Ray tried inputs like 1 100, 2 200, 3 300, and 4 400. He was shown the relationship between POLY and SPI by putting POLY 100, 200, and SPI 2 200, on the screen one after the other. (See Figure, 15.5)

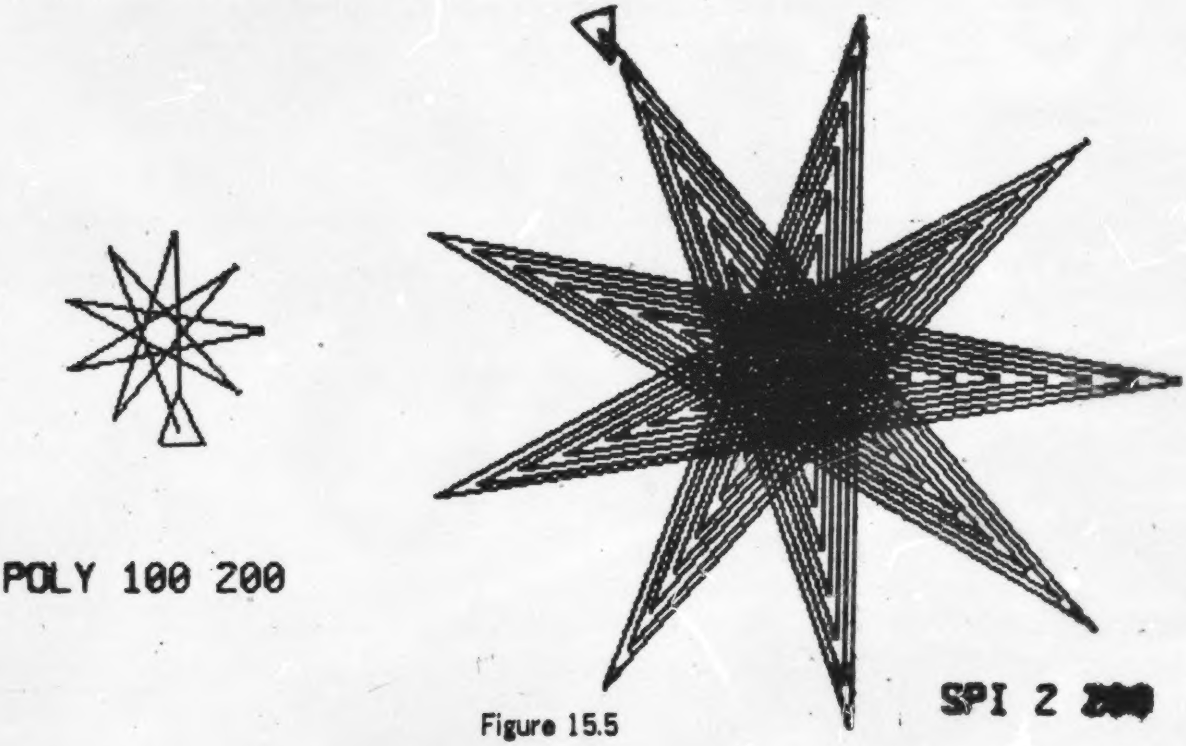
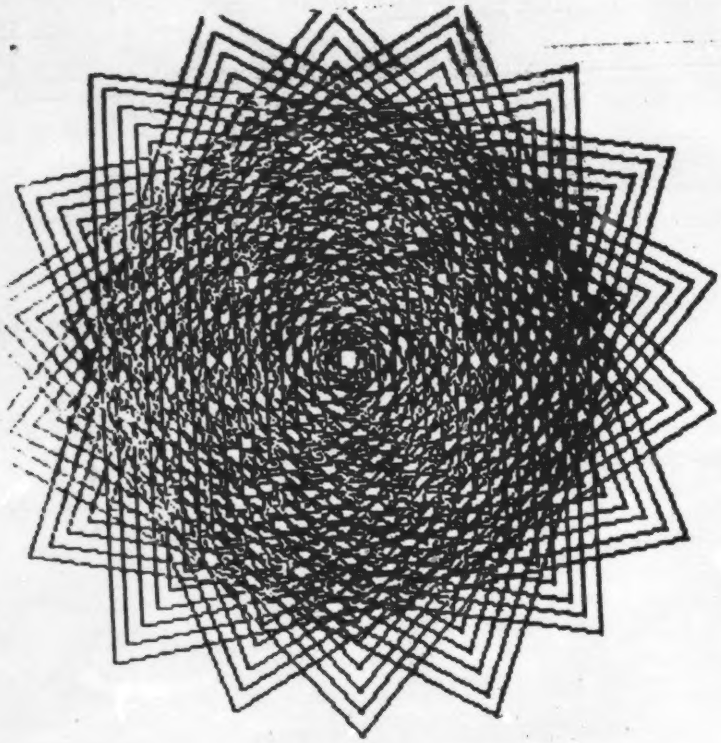


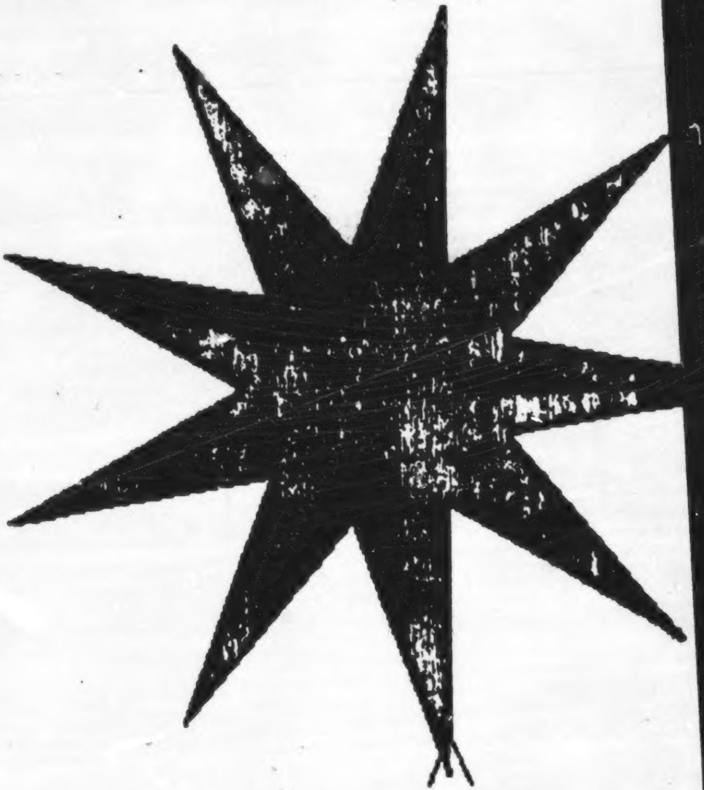
Figure 15.5

Ray was a lot more interested now, but he still felt that the spirals went off the screen too quickly. SPI was edited to allow changes in the increment, adding the variable GROW, and Ray was shown how to use it.

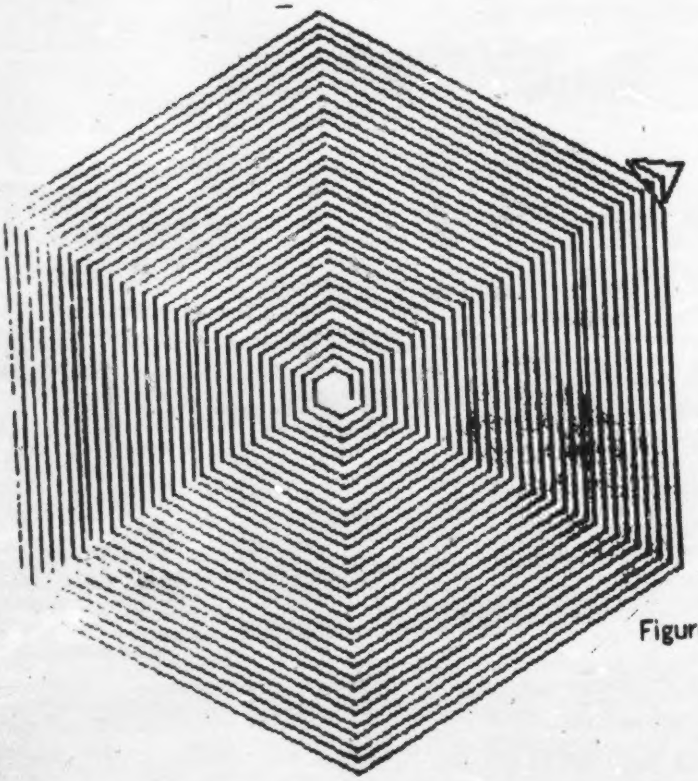
Ray liked the effect of the tighter spirals -- especially the emergent designs, which became apparent with small increments. He tried 1 400 1, 1 400 3, 1 401 1, focussing now on the effect of changing the numbers, rather than on the numbers themselves.



SPI 10 100 1



SPI 10 200 1



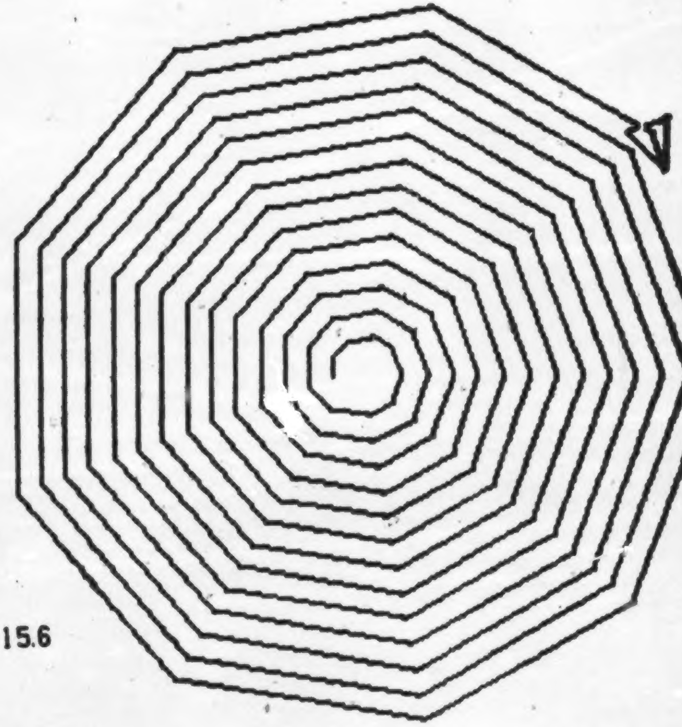


Figure 15.6

SPI 10 300 1

SPI 10 400 1

Ray came in with his own idea for the next class. "Can I put SPI and POLY together?" The teacher had also made up a worksheet for him which listed a few POLY and SPI designs, and left space for him to write down some "interesting" numbers of his own choosing. He worked with these activities for a solid hour, asking for help at only two points, writing down several "good numbers" on the chart he had been given. He liked one of the designs a lot, SPI 10, 150, 2, and called people over to see it. At the end of the class he spontaneously punched holes in his papers and put them carefully in his notebook. His explorations had been much more systematic: changing only one variable at a time.

POLY		SPI		
SIZE	ANGLE	SIZE	ANGLE	INCREMENT
50	100	10	30	2
50	35	10	45	2
50	49	10	150	2
		10	300	2
		10	100	1
		10	200	1
		10	300	1
		10	400	1
		10	500	1
	Ray's Favorite	10	150	1

(See figure, 15.6)

In the next session, Ray continued to work with the SPI procedure, this time, focussing on the emergent spirals. He had drawn SPI 1 250 2, and when the teacher suggested making a small change in the "middle number", he followed through by using 245, 235, 225 and 215, in sequence. He was also shown how varying the third input could create quite different effects, by decreasing the density of the design.

In this class, Ray began to use animation again as well. He was finally becoming comfortable with the computer. Although he was not defining procedures, he was engaging in significant mathematical explorations, and, best of all, feeling that he was in charge.

In the following session he continued to use right and left POLYs, some of his favorite SPIs, and went back to animation. He learned that he could animate anything, using his NKP procedure, just by typing MAKE "FB SNAP, and then typing NKP -- which would animate whatever had been on the screen. He also asked the teacher to write down the WRITE command in his notebook.

At the end of the period, he came over to where Gary was working and asked what he was doing. This was the first time he had taken an obvious interest in anyone else's work. In addition, he very carefully made sure to remind Gary: "better make sure to write your file before you say GOODBYE" thus letting it be known to one and all that "he knew what it was all about" as well as anyone.

The next class was "visiting day" and Ray brought Paul, a seventh grader. He showed Paul how to use POLY and SPI procedures and a few other procedures: NKP, SAM and RG. Together they tried out different inputs to POLY and SPI. Ray referred to his notebook for ideas about what inputs to try, and wrote down new ideas as he went along. Ray stuck strictly to what he was comfortable with: POLYs, SPIs and moving his rocket. He and Paul had a wonderful time, and both came away feeling wonderful.

4. Conclusion

This turned out to be Ray's last class. He truly went out in a "blaze of glory". If we had understood earlier how afraid of failure Ray was, and how important it was for him to have only one thing to vary at a time, we might have been able to get him "hooked" much earlier. Although he did not define any more procedures of his own, or do any "planning and debugging", Ray had finally achieved the first prerequisite for any success with the computer. He had found a way to be in charge.

One last "footnote" about Ray. When the classroom teachers were interviewed, they felt that the computer experience had had a profound effect on Ray. One of his teachers put it this way:

> "There was a breakthrough with Ray...He hasn't connected all year...been floating, not that there is any resistance or hostility, but just no connection...(he) was probably swamped by the reading required this year...The breakthrough for him in LOGO, the success he has had, is powerful information for me...he has produced the best piece of writing I've seen from him...His physical arrangement has changed, he was isolated in the room before, now he sits with others."

16. Tina

Tina is a student who has great difficulty with academic work. In mathematics and language she is considered to be several years behind her classmates (her national achievement test scores place her in the 3nd percentile). Although she receives several hours of special tutoring in the school's Learning Center each week, the staff is concerned about her rate of progress, and are struggling to find an appropriate educational program for her as she enters seventh grade.

From the beginning of her LOGO classes, Tina's work provided an exception to most of the patterns we have observed among the rest of the students in our study. Although Tina did very little Turtle Geometry, she was more "involved" with the computer than any other student. She developed an almost personal relationship with the computer she used, giving it a name, and treating it in the way some chidren treat a doll or a pet.

Tina's major use of the computer was as a typewriter and editor. Using a text-editing program designed to allow her to use the printing capability of the computer without writing procedures or using PRINT commands, Tina wrote a series of nine "stories", which she distributed to her family, friends and teachers. These stories constituted the most significant work of Tina's entire school year.

Tina's success in writing and completing work with the computer has led to success in other areas as well. Near the end of sixth grade Tina began turning in assignments on a regular basis -- something she had not been able to do before taking the computer classes. In addition, a friendship established with Harriet, another girl in her LOGO class, has led to Tina's being included socially with Harriet and her friends in the classroom, ending her previous isolation.

16.1 Tina's "Personal Relationship" with the Computer

Tina had a very special relationship with one of the computers, which she called "Peter". While this was most noticeable in the earlier classes, it persisted throughout Tina's computer experience. In the first class, Tina expressed an interest in communicating with the computer: "I thought it would talk to you. Hello." When it responded to a missing space in an FD445 command by printing "You haven't told me how to FD445," Tina responded with genuine anger: "I did, you fool!" "Oh, this isn't doing what I told it to do," she said at another point.

In the next class she again asked about communication: "I want to ask it questions. Oh you know, What's your name? Things like that." The teacher explained that all the computer's answers would have to come from her, and showed her how to use a PRINT command. Her first procedure gave the computer an answer to her first question:

```
TO WHO
1 PRINT [MY NAME IS PETER]
END
```

Tina was extremely possessive about the particular machine she regularly used. She felt that this machine, her "Peter", was hers, and she would say things like "Take good care of Peter," when she left for the day. When she came late to the fourth class, "Peter" was already being used by another student. Tina stormed and fussed. "I want to use my computer. That's the one I use every day." When the teacher insisted they were all the same, Tina responded; "I hate you...I think I'm going to cry..." Finally she was persuaded to use another computer and was somewhat reassured by the fact that it responded correctly to WHO, once she had logged in. Although she actually worked quite successfully the other students all left "Tina's computer" to her in succeeding classes.

As the classes went on, Tina's relationship with "Peter" tempered somewhat. The teacher continued to stress that the computer was a machine, controlled by her and other programmers. Gradually she reduced her expression of anger at the computer, as she realized how error messages were related to what she had done, although she continued to share credit for her work with "Peter". "Do you want to see the stories me and Peter have written?" she would ask a visitor. And she would often introduce visitors to "Peter", using her WHO procedure.

By the time the classes ended, Tina had progressed to a more balanced understanding of her role in relation to the computer. Her behavior with the computer had become much more matter-of-fact and she had begun to interact very extensively with the other children in the class. As she came to understand the mechanical predictability with which the computer responded to her, and as she began to take pride in her accomplishments in story writing, "Peter" came more and more to take on the status of a personal fantasy -- one which a child knows is a fantasy, but persists in "playing" sometimes because it's fun.

16.2 Tina's way of working with the Computer

The strongest characteristic of Tina's approach to working with the computer was her need to clear the text display screen. While many children have a desire to clear the screen between projects, or when bored or frustrated, Tina exhibited this desire in an extreme way. One of Tina's first questions was about "how to get the words off the screen," and learning how to accomplish this using the carriage return was a major discovery of hers during the first class. Once she figured out how to do it she cleared the text display after almost every line -- especially after error messages. In later classes Tina developed the habit of clearing the text display before logging in, after logging in, after almost every instruction that was not part of an ongoing task, and before and after saying GOODBYE.

Tina's habit of clearing the text display in this way interferred significantly with her learning. It effectively prevented her from maintaining continuity in what she was doing, and eliminated the possibility of her responding to the computer's error messages. Even after completing a piece of work Tina would hardly look at it: after she had edited her first letter, DORIS, she swept it off the screen without reading it.

In her pre-LOGO interview, Tina used a similar strategy when trying to arrange a series of permutations of a set of colored blocks. The task was to create as many different arrangements of the blocks as possible.

After having done a few, Tina swept the old ones away, clearing the table -- thus making it impossible for her to know whether the next one was different or not. When the interviewer re-established all Tina's old arrangements, she made one more, and then compulsively swept them all away again.

Tina eventually learned to leave several sequential lines on the screen, while working on a specific project -- although it required a definite effort on her part not to endulge her carriage return habit. When working on a drawing project, she would carefully copy all the necessary steps into her notebook, then quickly clear the screen.

Tina had a strong desire for completion and correctness in all her work. Although she was quite a perfectionist by her own standards, sometimes working a whole hour on one story, Tina never proofread, edited or added to a story once she had declared it finished. Unless she noticed an error in a particular line while typing it, or immediately after typing a carriage return, she never went back to find an error. Once a story was done, it was done! She might show it to other people, but she rarely read it over to herself, and certainly not with a view to analyze or correct it. Incidents like the following were observed several times. Tina was proofreading a sentence which began "ONCE UPON TIME..." but she read aloud, "Once upon a time..." and continued, satisfied that her work was correct.

Tina did develop one way of editing stories. If she made a mistake early in a story she could start the entire process again by typing END, and clearing the screen. Typing LET LETTER again would clear out her workspace and allow her to begin her story again. When she did this, she almost always typed an identical sentence to the one she had typed before -- without any printed reference -- sometimes complete with the error that had forced her to start over in the first place. In such case she would stop and go through the whole process again until she got it right.

(Tina's attitude towards errors and towards completion of "correct" work is also shown by her behavior in other situations. One of her learning disability tutors related that she would repeatedly throw out whole pages once she had made one error -- a tactic that frustrated her in working with a standard typewriter. Hence in her written work she had developed a style of writing only short pieces of one or two "perfect" sentences.)

Tina also had a great desire for neatness in the classroom. She took it upon herself to collect and put away pens, notebooks, and other items which had been left out by the other children, and to remind them to put their things away.

In her work with the computer, Tina established a set of conditions about what she would and wouldn't do, and adhered to these with little change throughout the classes. Tina would do the following: LOGIN "TINA; LET LETTER (to get her letter writing procedure); start, write and end a letter; run a letter or a drawing procedure by typing its name; and type the word GOODBYE. Things she wouldn't do on her own included: writing a procedure; editing a procedure or a story; saving a procedure, by writing a file; reading a file; and using the lineprinter. These parts of her daily routine were done for her by her teacher.

It was very important for Tina to keep a firm boundary between what she would and wouldn't do. Towards the end of the classes, an effort was made to get her to use the lineprinter on her own, showing her the PRINTON command, and showing her how to make multiple copies of a story by using REPEAT. At first Tina objected to this strongly, but eventually agreed to do some of it. With stronger insistence and more time, Tina might have learned these commands, and attained a greater measure of independence in her computer use.

Tina took a large part of each class to settle down to productive work. She usually came to class later than the other students, and wandered around a bit before deciding what to do. Once started on a story, she often made a lot of mistakes on the first line -- requiring her to start the whole story over again. Once past the first line, however, she typically finished a story without any more false starts. Her problem with starting was reflected in a rush at the end of the class. Since Tina placed a premium on completed work, she often found it necessary to stay after class to finish up.

16.3. Tina's Use of the Turtle

Tina used the TURTLE in only nine of the twenty-four class sessions. Most of her use of the TURTLE required active help from the teacher or one of the other students.

From the beginning, Tina had great difficulty concentrating on what the TURTLE was doing in response to the commands she typed. In the first class she literally "dozed off" while the rest of the class was figuring out how to draw a box, and how to write a procedure. When she began to work on her own, she worked randomly, copying commands from an instruction sheet without looking at their effect, and clearing the actual commands from the text display almost as quickly as she typed them. For numerical inputs she chose double numbers like 55, sequences like 34, or combinations like 445. At the end of the period she wrote a few steps and a comment in her notebook:

1) ? F.D. 45
? B.K. 23

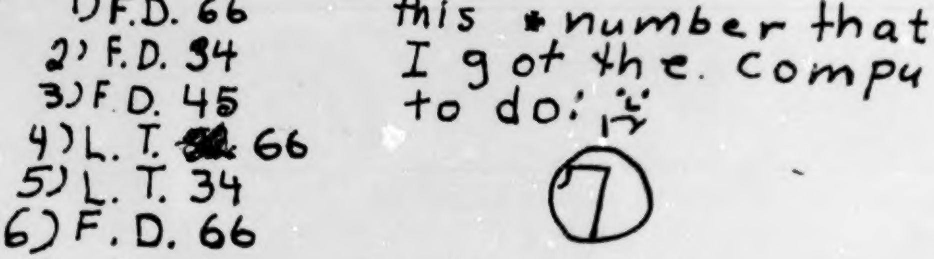
3? F.D.23
? Rt 34

I told Computer to go Leftand:: right and FORWAWD

Figure 16.1

Although Tina seemed to have difficulty throughout the first class in focussing on what the turtle was doing in response to her typing, she did notice when the command HT 78 made the TURTLE disappear, and asked for help saying "I want my TURTLE back." She had a much stronger response to the computer's text responses -- angrily commenting about error messages and sweeping them off the screen with a stream of carriage returns.

In the second class she was more thoughtful about what she was doing, eventually noticing that she had caused the computer to draw a "seven", and carefully copying down the steps in her notebook:



1) F.D. 66
2) F.D. 34
3) F.D. 45
4) L.T. 66
5) L.T. 34
6) F.D. 66

this number that
I got the compu
to do:

Figure 16.2

With much help, this was made into the procedure SEVEN, copying the steps from her notebook, (fig. 16.3).

```
TO SEVEN
1 FORWARD 66
2 FORWARD 34
3 FORWARD 45
4 LEFT 66
5 LEFT 34
6 FORWARD 66
33 FORWARD 45
333
END
```

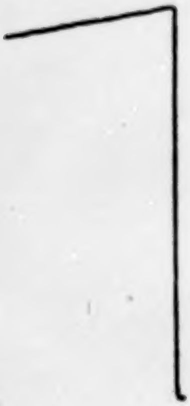


Figure 16.3

In another class Tina decided in advance that she wanted the computer to draw a "zero". When asked to draw a zero by hand, she drew it as a rectangle, possibly indicating that she understood that it was easier for the TURTLE to draw straight lines than to draw curves. She required a great deal of help in building her rectangle with the TURTLE. Tina could not determine the angle to use at each corner of the rectangle, although she did know that opposite sides had to have the same length. She decided to call her rectangle TINA and carefully copied the steps into her notebook.

When she taught her procedure to the computer, Tina had difficulty copying the steps. When she typed 66 as a line number instead of 6, Tina insisted on redoing the entire procedure instead of editing. When the same thing happened again, she allowed the teacher to fix it, by erasing the line for her.

```
TO TINA
1 FORWARD 66
2 FORWARD 45
3 RIGHT 90
4 FORWARD 23
5 FORWARD 45
6 RIGHT 90
7 FORWARD 66
8 FORWARD 45
9 RIGHT 90
10 FORWARD 23
11 FORWARD 45
```

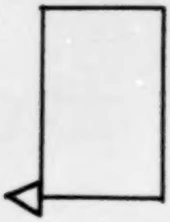


Figure 16.4

With Harriet's help, Tina then spent some time repeating her design, and making it spin. She never returned to Turtle Geometry activities on her own.

If Tina had not had the possibility of using the computer to write stories, she might have persevered with drawing a lot more. Her experience with the "zero" drawing shows that she was <u>capable</u> of seeing the connection between the commands she was giving and what the TURTLE was drawing. On the other hand, Tina was delighted with using the computer to write stories. This was a unique and special use of the computer, and she did not have to compare her work with anyone else's. Drawing with the computer may have seemed to be just another activity at which she was "not as good" as her classmates.

16.4 Tina's Stories

Most of Tina's time and effort with the computer was devoted to writing. A special program, LETTER was created for her use, allowing her to type a story directly into the computer without having to write a procedure, or use line numbers or PRINT commands. Tina used the LETTER procedure to write two "letters" and seven "stories" over a period spanning twenty classes. She also started at least three more stories that were discarded before she finished them.

In her stories, Tina had a product of which she could be proud. She could easily share copies with her classmates, teachers and family and with visitors to the LOGO classes. It was a unique product which could not be compared with the work of other students. This was very important to Tina, who was used to having her work evaluated as being of lower quality than that of her fellow students. Most importantly, Tina was able to use the computer to express her ideas in writing and to improve her writing skills in a way that she had not been able to do in regular classes or in her special tutorial programs.

Here are Tina's nine "stories" (as she called them), beginning with DORIS (class 4) and continuing to MISS.HIRSH (class 21). They show considerable development in style, sophistication and in proficiency in grammar and punctuation. They are presented here exactly as Tina wrote them.

DORIS

DEAR DORIS HOW ARE YOU DOING UP IN NEW YORK. I HOPE I WILL SEE YOU IN THE SUMMER
LOVE TINA AND PETER

HELEN

DEAR HELEN HOW ARE YOU IN YOUR NEW HOME. I AM GOING TO GET YOU SOMETHING FOR YOUR NEW HOME .AS SOON AS I GET MY MONEY
LOVE TINA

TOMMY

TOMMY IS LITTLE BOY HE LIVE' IN JAMES TOWN FOR 3 YEARS HIS MOTHER
DIE WHEN HIS FATHER CAME BAKE ARMY 2 YEARS AFTER THAT THEY MOVE
TO NEW YORK AND HIS FATHER GOT MARRIED THEY HAD A LITTLE GIRL
NAME LIZ.14 YEARS AFTER THAT TOMMY GOT MARRIED TO A TEACHER
ENGLISH HIGH AFTER THEY GOT MARRIED TH HAD TO BABY'
LITTLE BOY TOMMY JR AND LITTLE GIRL NAME LISA AFTER
THE CHILD WERE 3 AND 2 THEY MOVE TO HOLLY WOOD

ANN

ANN IS A OLD WOMAN SHE IS 81 YEAS OLD
THEY H BUSTER THAT IS MARRIED TO
A NURSE AND THEY HA TRIPLETS. THEY BE MARRIED FOR
3 YEARS AFTER THEY GOT SETTLED DOWN THAN THEY
MOVE TO NEW YORK WERE DORIS LIVES. ABOUT 5 YEARS AFTER
ABOUT 5 YEARS THEY HAD A LITTLE GIRL SUE AND WAS PRETTIEST

SONNY

SONNY IS A LITTLE BOY HE LIVES WITH HIS AUNT HELEN IN CALIFORNIA HE HAS BEEN LIVE WITH HER FOR 9 YEARS. HE IS GOING TO A HOME FOR LITTLE WONDERS 4 WEEKS AFTER THAT TO COUPLE A ADOPTED SONNY HE WAS THE HAPPIEST BOY THAT YOU EVERY SEEN. I GUESS IF THAT WAS ME I WOULD BE HAPPY IF SOME ONE WOULD ADOPT. BUT SEE I AM NOT ADOPT I HAVE MY ON MOTHER AND I AM GLAD THAT I HAVE MY ON MOTHER.BECAUSE THE KIDS THAT HAVE FEELS REALLY BAD. THAT'S WHY ALL THE KIDS IN THE WORLD SHOULD BE GRATEFUL TO THEIR PARENTS. THE END.

DONELL

DONELL IS A GROWN MAN HE LIVES WITH HIS FRIEND IN NEWYORK HE BE LIVE THERE FOR 3WEEK'S THEY BOTH BEING LIVE THERE FOR 4 WEEKS. THEY BOTH HELP PAY THE RENT THEY PAY 200 DOLLAR A WEEK FOR RENT JIM GUESS WHAT ?WHAT I GOT FIRE TODAY WHAT DID YOU SAY I GOT FIRE TODAY.WHAT ARE WE GOING TO TELL MR. SMITH I DO NOT KNOW WHAT TO TELL MR. SMITH ABOUT THE RENT.

HARRIET

HARRIET IS A LITTLE GIRL SHE LIVES OUT IN BROOKLINE WITH HER MOTHERABIGAIL AND HER FATHER EGBERT.SHE BEING OUT IN BROOKLINE FOR 11 YEARS.SHE GOSE TO A SCHOOL CALL WILLIAM H. LINCOLN.SHE BEING GOING THERE FOR 6 YEARS AFTER 6 YEARS SHE START GOING WITH A BOY NAME DENNIS COOPER FOR 3 MONTHS.

```
            MR. LEWIS

ONCE UPON A TIME THERE WAS A MAN NAMED BOB .HE LIVED
OUT IN BROOKLINE HE LIVED OUT THERE FOR 10 YEARS.
2 YEARS LATER HE FOUND A JOB OUT IN NEWTON .NEWTON
WASNT PAYING ENOUGH MONEY TO SUPORT HIM .HIS FRIEND
HAD TOLD HIM TO GO SEE DAN WILSON. THE NEXT DAY BOB WENT OUT TO SEE MR.
WILSON. MR. WILSON SAID BOB WHAT KIND OF WORK CAN YOU DO. I CAN TEACH
.I CAN DO PAINTINGS DRAWING AND ALL KINDS OF THINGS.WHAT WOULD YOU LIKE
TO DO MOST. I WOULD LIKE TO TEACH. I THINK KIDS NEED MY HELP. WHAT
KINDS OF THINGS WOUULD YOU LIKE TO TEACH. IF I MAY I WOULD LIKE TO TEACH
 MATH, AND SCIANCE, AND ENGLISH. ONE WEEK AFTER
HE START WORK FOR LINCOLN
BY TINA DEBORAH AND PETER

            MISS.HIRSH

ONCE UPON TIME THERE WAS A LITTLE GIRL NAME LISA SHE LIVED IN
BROOKLINE. SHE LIVE THERE FOR 16 YEARS WITH HER MOTHER HELEN
WAS THE BEST WOMAN IN THE WORLD .  THAT IS WHAT HER LITTLE
LISA THOUGHT OR HER MOTHER HELEN .SEE HELEN WAS NOTHING BUT
A NASTY OLD 42 YEAR WOMAN.SHE DIDN'T CARE ABOUT LISA WENT TO
SCHOOL VERY NASTY.SHE LOOK LIKE SHE HAVEN'T TOOK A BATH IN
6 MONTHS.AND ROLL THROUGH THE MUD.
BY TINA TO LISA HIRSH
```

Tina's first two stories, DORIS and HELEN, were really letters, written to her godmother and mother respectively. The next four were all "made up stories about real people." Tommy, Ann, Sonny and Donell are all Tina's relatives. The last three stories, HARRIET, MR.LEWIS and MISS.HIRSH were about a classmate, and about Tina's regular sixth grade teachers.

In the first few classes, Tina had shown an interest in "communicating" with the computer, and had been shown how to write procedures using PRINT commands.

Starting with the fourth class a special program was created for her to allow her to type whatever she wanted directly into the computer. The procedure, LETTER, required that Tina choose a name for a story, and then it interpreted each of her typed lines as a separate REQUEST. If Tina wanted to change a line she could use the rubout key, before she typed a carriage return. Other than that, LETTER had no special editing capability. When Tina finished a story, she typed END, which caused the LETTER procedure to finish defining the procedure she had named. It then typed out, "HERE IS YOUR LETTER, DORIS" (for example), and ran the procedure, printing the letter exactly as Tina typed it.

Tina was quickly able to use the LETTER procedure on her own. She would type LETTER, choose a name for her story, and begin. If she wanted to change a line after typing a carriage return she had to be helped by the teacher. Sometimes she accidentally typed a carriage return intending to type a rubout. After typing a string of carriage returns, she would often type END and start all over. This was Tina's only way of independently editing a story procedure.

Tina was introduced to LETTER in class four -- the same class in which she made such a fuss about having to use a "different" computer. Her letter, DORIS, was first written out by hand, and then copied on the computer. At the end of the class several copies were printed out for her, as she carefully estimated how many she would need: "One for me, one for my mother, one for Doris (her godmother)," etc. In her notebook, Tina wrote. "Today me and Peter did a letter".

Tina's next letter was HELEN, written during class six. This letter, accomplished in a much shorter time, was written to her mother. It followed the same format as DORIS: greeting, question, statement, signature. Punctuation consisted of two periods.

In class eight, Tina wrote a "story" she called TOMMY. After several false starts she settled down to complete a seven line story, which ended rather abruptly. She had trouble typing the first line and used a lot of extra carriage returns between lines. Twice she was helped to erase the story and start again. Once she completed the first line successfully, she went on to write seven lines, before deciding she was finished.

The story TOMMY represented a big change from Tina's previous computer writing, and set the tone for the rest of the writing she was to do. In TOMMY, her concern was not for a grammatically correct careful statement, following a prearranged format. Rather she was concerned with the details of the story: the names of the characters, the places they lived, the sequence and timing of events. She was reminded not to type a carriage return until she was satisfied with an entire line, and she often rubbed out all or most of a line to correct a spacing or spelling error. Once a line was finished, however, she did not ask for corrections.

Tina's next story, ANN, ended even more abruptly. In this case, time ran out before she completed it, and Tina showed no interest in adding to it at a later class. She worked very carefully on her first seven sentences, rubbing out and rewriting as often as necessary, carefully checking each sentence before going on.

She finally got frustrated in starting the eighth line, typed a lot of carriage returns and called for help. After her mistakes had been corrected Tina declared the story finished, and printed out about 10 copies, offering them to classmates as well as to two visitors. She appeared to have a great deal of pride in her work.

Tina's next story, SONNY was her longest to date -- nine lines. In writing the story Tina was interested in punctuation, checking with the teacher to see if each period was located correctly. At one point Tina asked whether she should use "adopt" or "adopted" as a spelling. The difference in meaning was explained by using each in a sentence.. Tina seemed to be unable to hear the difference in the way the words sounded, so that the choice was completely arbitrary to her. She decided to use "adopt" at that point, and throughout the rest of the story.

Tina was clearly using this story to express some very deep feelings and values -- as well as some appreciation for her own parents.

After Tina had completed SONNY, she was asked for some samples of the writing she was doing in class. "I don't have time to write stories in class," was her reply. "I've got too much work to do." Tina's English teacher and her learning disabilities teacher both confirmed the fact that Tina had done virtually no creative writing this year. Her English teacher explained that Tina rarely completed any work -- pointing out that Tina's computer stories were among her first finished pieces of work.

Tina's next story, DONELL seemed to have the making of a real story with a definite plot line, and realistic dialogue. Once again, Tina had difficulty settling down to start her story, and ran out of time before finishing it.

After a lot of disruptive fooling around with the other kids, she settled down to write her story, HARRIET, which was well constructed and carefully written. She paid careful attention to spelling and punctuation -- asking for assurances on the placing of periods at the ends of sentences.

In class 20, Tina dictated a story to her friend, Deborah, who had come as a visitor. The resulting story, MR.LEWIS was Tina's longest and clearest story, and had a more direct flow than usual in Tina's stories. Deborah typed the story exactly as Tina dictated -- Tina dictated the puncutation as well. Her verbal dictation seems to have been closer to standard english grammar than her own typing. The story had a clear plot, with a definite beginning and ending. Tina ended the story by writing "BY TINA DEBORAH AND PETER," after thinking for a few minutes about how to share the credit.

Tina's last story, MISS.HIRSH, was written in only 20 minutes. She typed without any significant errors or false starts. She was particularlly concerned about the location of periods, and the use of apostrophes in DIDN'T and HAVEN'T. (She knew where the apostropes went, but wanted to have that confirmed by the teacher.) After finishing the story, Tina said with a gleam in her eye "Miss Hirsh can take a joke."

16.5 Some Possible Reasons for Tina's Success in Writing

In a meeting held after the end of computer classes, Tina's English teacher said that Tina had begun making a significant effort to complete assignments and turn in work. She said that Tina's computer stories were the first completed work that Tina had turned in this year.

After consulting with Tina's teachers, some tentative insights can be offered about why Tina was able to accomplish so much writing in the LOGO classes, while she has been unable to accomplish much writing in either her regular classes or with her language tutors.

Tina is a fanatic about completing work without errors. When an error occurs in Tina's normal work, she tears up the paper and throws it away. Since she never examines her errors, or tries to learn from them, errors are seen by her as a total disaster. Using the computer, she was able to rubout many errors as they occurred without destroying all her work up to that point. In this way she could be satisfied with her finished product. Once she declared herself "finished" she never looked back to see if there were any other errors.

The printed computer output had a "professional" quality that looked more polished than other children's writing. In addition, she could have as many copies as she wanted at no extra effort -- which could bring her positive attention from friends, family, and several different teachers, all in the same day.

Tina seems to feel extremely sensitive to any "correction" of her work whether direct or implied. This is confirmed by her abhorance of "errors". She was not corrected or criticised while writing, and the teacher's assistance was confined to answering her questions about spelling and punctuation. Therefore, since Tina had worked hard to make her work "correct" and since it looked "professional," Tina assumed that it was correct, and was very proud of it.

Tina knew that her work was unique among the kids in the computer classes. Just as only Tina used the computer she called "PETER", only Tina could use the LETTER procedure to write stories. Thus Tina could feel that her work was at least as good as the other students' work. Tina has rarely had the opportunity to feel that way about her work.

16.6. Tina's Social Behavior During LOGO Classes

Just as Tina's "relationship" with the computer was of the utmost importance to her, she also placed a high premium on her personal relationships with the people around her. At first Tina had a great deal of sensitivity about being observed by adults in the classroom. She was particularly concerned about observers who took notes, and she would occasionally ask them what they were writing about. Tina made a point of introducing herself to all the visitors who came at different times to observe the classes, and of establishing a definite relationship with any who came more than once.

Tina insisted that she be noticed. When she came to class late, she would often sing or shout loudly in a powerful voice as she walked in the door. If a visitor was present, however, she would quiet down suddenly in embarrassment. During any class she might call loudly across the room to a teacher, visitor or a classmate.

At times, Tina could be quite manipulative -- especially when she wanted help. She could be loud and insistent calling across the room for assistance, and a few minutes later rebuke another child for "interrupting". She often corrected other people's behavior, saying that some action, or omission was "rude", or "sloppy". This was usually done in a somewhat cheerful, "motherly" tone that had only a slight undercurrent of hostility.

In her relations with her classmates, Tina adopted a "breezy," somewhat teasing style, characterized by a cheerful good humor. Occasionally this "breeziness" got out of control, and became disruptive and rowdy, including chases around the room, hitting and hair-pulling. At such times, Tina had a very difficult time settling back into appropriate class behavior.

At other times -- especially with visitors present -- Tina would be extremely quiet, polite and demure. It was as if she had two sets of rules -- one for serious behavior in public -- the other for playful behavior in private. As the classes went on, it was clear that Tina considered the LOGO classroom as a "private" place, where she was among friends and could be her playful self.

At first Tina was tolerated, somewhat uneasily by the other kids. Gradually, Harriet and Dennis became quite involved with her playfulness, and cheerful bantering. Most of the time they seemed to enjoy the distraction she provided, and allowed her to provide an excuse for letting their own playfulness emerge on occasion.

Harriet became quite protective of Tina, sticking up for her on occasion, as well as helping her with her work from time to time. Tina consulted Harriet frequently about spelling and punctuation. Harriet spent a lot of one class period helping Tina type a TURTLE procedure, and showing her how to use the SPIN command.

Tina took a lot of interest in Harriet and Dennis as a "couple", teasing them regularly about their friendship. She wrote their "relationship" into her story, HARRIET, and spent a good deal of one class period watching them play Dynaturtle games together -- commenting regularly, not on their dynamic strategies, but on their personal interchanges.

Harriet's interest in Tina was evidenced outside of LOGO classes as well. One of Tina's classroom teachers reported that she had moved her seat in class to be near Harriet and her group of friends. In becoming accepted by them, she had moved from virtual "isolation into the core of the class, socially." Her teacher attributed this to the relationship Tina developed during the LOGO classes and to the feeling of success she has had with her computer work.